

## Logic circuit arrangement

Publication number: DE10354501

Publication date: 2005-06-30

Inventor: GLIESE JOERG (DE); SCHEPPLER MICHAEL (DE)

Applicant: INFINEON TECHNOLOGIES AG (DE)

Classification:


- international: **H03K19/173**; H03K19/173; (IPC1-7): H03K19/173;  
H03K19/094

- European: H03K19/173C2

Application number: DE20031054501 20031121

Priority number(s): DE20031054501 20031121

Also published as:

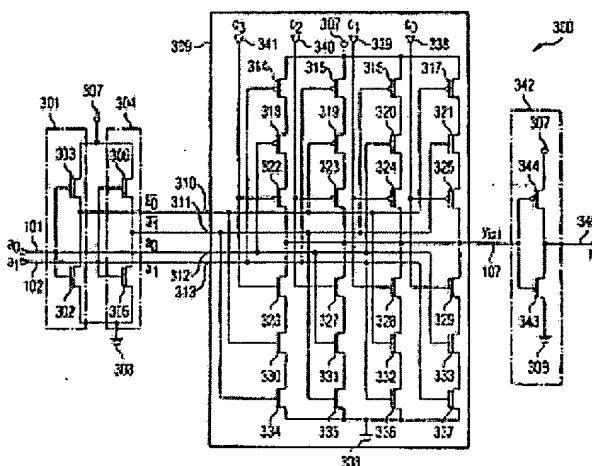
 US2005134317 (A1)

Report a data error here

Abstract not available for DE10354501

Abstract of corresponding document: **US2005134317**

A logic circuit arrangement contains at least two data signal inputs, at which at least two data signals can be provided, and contains a first signal path having a plurality of transistors of a first conduction type, said signal path being coupled to the data signal inputs. The logic circuit arrangement furthermore contains a plurality of control inputs coupled to the transistors.



Data supplied from the **esp@cenet** database - Worldwide



(19)  
Bundesrepublik Deutschland  
Deutsches Patent- und Markenamt

(10) DE 103 54 501 A1 2005.06.30

(12)

## Offenlegungsschrift

(21) Aktenzeichen: 103 54 501.8

(22) Anmeldetag: 21.11.2003

(43) Offenlegungstag: 30.06.2005

(51) Int Cl.<sup>7</sup>: H03K 19/173  
H03K 19/094

(71) Anmelder:  
Infineon Technologies AG, 81669 München, DE

(74) Vertreter:  
Viering, Jentschura & Partner, 80538 München

(72) Erfinder:  
Gliese, Jörg, 80469 München, DE; Scheppler,  
Michael, 82194 Gröbenzell, DE

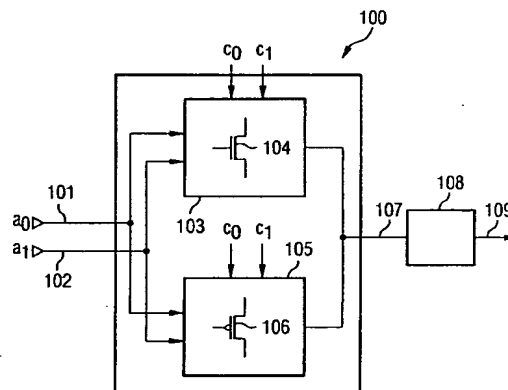
(56) Für die Beurteilung der Patentfähigkeit in Betracht  
gezogene Druckschriften:  
DE 31 48 410 C2  
US 65 29 040 B1  
US 62 85 218 B1  
EP 05 73 175 A2  
WANNEMACHER, Markus: Das FPGA-Kochbuch,  
Bonn:  
International Thomson Publishing Company,  
1998,  
S. 111 u. 197;

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

Prüfungsantrag gemäß § 44 PatG ist gestellt.

(54) Bezeichnung: Logik-Schaltkreis-Anordnung

(57) Zusammenfassung: Eine Logik-Schaltkreis-Anordnung enthält mindestens zwei Datensignal-Eingänge, an denen mindestens zwei Datensignale bereitstellbar sind, und enthält einen mit den Datensignal-Eingängen gekoppelten ersten Signalpfad mit einer Mehrzahl von Transistoren eines ersten Leitungstyps. Die Logik-Schaltkreis-Anordnung enthält ferner eine Mehrzahl von mit den Transistoren gekoppelten Steuereingängen.



**Beschreibung**

**[0001]** Die Erfindung betrifft eine Logik-Schaltkreis-Anordnung.

**[0002]** Mit dem Aufkommen der Digitaltechnik und der sich sprunghaft entwickelnden Mikroprozessortechnik entstand ein Bedarf nach programmierbarer Logik. PLDs ("Programmable Logical Devices") sind integrierte Schaltkreise, die vom Anwender mittels Programmierens in ihrer Logik-Funktion festgelegt werden. Ein PLD ist eine regulär aufgebaute Architektur für digitale Logik-Operationen mit einer Vielzahl von Schaltern, die eine Vielzahl von Signalpfaden ermöglichen. Die einem PLD anwendungsspezifisch zugeordnete Logik-Funktion wird durch die Konfigurierung des PLDs festgelegt.

**Stand der Technik**

**[0003]** Zu PLDs gehören unter anderem Field-Programmable Gate-Arrays (FPGA), deren Funktionalität ihnen vom Anwender zugeordnet werden kann, Mask Programmable Gate-Arrays (MPGA, auch "structured ASICs" genannt), welchen mittels hardwaremäßigen Konfigurierens eine Logikfunktion zugewiesen werden kann. Via Programmable Gate Arrays (VPGAs) gehören zu den MPGAs.

**[0004]** Eine digitale Logikzelle bildet  $n$  Eingangssignale auf ein Ausgangssignal ab. Die Anzahl der möglichen Abbildungsfunktionen ist

$$2^{2^n}.$$

Eine solche Schaltungsgruppe wird gemäß dem Stand der Technik zum Beispiel unter Verwendung von sogenannten Look-up-Tabellen (LUT), anschaulich Nachschlagetabellen, realisiert. Hierfür werden Funktionswerte der Logikfunktion mittels eines Datenwortes von  $2^n$  Bit eingestellt. Mit anderen Worten ist die jeweils ausgewählte Logikfunktion in dem Datenwort kodiert. Entsprechend der ausgewählten Logikfunktion werden die  $n$  Eingangssignale  $a_0, a_1, \dots, a_{n-1}$  miteinander verknüpft. Somit können die Logik-Eingangssignale der Logikfunktion  $y = f(a_0, a_1, \dots, a_{n-1})$  als binäre Adresse angesehen werden. Die Logik-Eingangssignale werden in eine sogenannte One-Hot-Codierung gewandelt, anschließend wird über eine Pass-Gate-Logik ein Funktionswert gewählt. Ein derartiges Verfahren ist zum Beispiel aus [1] bekannt.

**[0005]** Gemäß [2] können die Eingänge als Steuereingänge für einen Multiplexer-Baum dienen. Die Multiplexer können logikbasiert und/oder auf Basis von Transmission-Gates realisiert werden.

**[0006]** In [3] ist ein FPGA auf Basis einer Look-Up-Tabelle (LUT) offenbart. Gemäß [3] wird eine Logikfunktion erst durch Kombination des Wertes von Eingabesignalen (IN1, IN2, ...) mit Logikauswahlsignalen (Q1, Q2, ...) ausgewählt. Anders ausgedrückt kann gemäß [3] mittels Anlegens der Eingabesignale und der Logikauswahlsignale an Transistor-Eingänge eines Multiplexers eine Logikfunktion ausgewählt werden.

**[0007]** Die aus dem Stand der Technik bekannten Lösungen unter Verwendung einer Look-Up-Tabelle weisen hinsichtlich Schaltgeschwindigkeit bzw. Störsicherheit Nachteile auf. Die bekannten Lösungen lassen sich ferner für viele Anwendungen nicht ausreichend kompakt in Layout realisieren. Daher ist mit den aus dem Stand der Technik bekannten LUT-Lösungen eine fortgesetzte Skalierung nur schwierig möglich.

**[0008]** Alternativ zu den bekannten LUT-Architekturen sind aus dem Stand der Technik Verschaltungen aus einzelnen Logikgattern bekannt, mit denen eine gewünschte Logikfunktion aufgebaut werden kann. Allerdings ist eine solche Architektur auf das Bilden einer ganz bestimmten Logikfunktion beschränkt, wohingegen der Gesamtumfang aller möglichen Logik-Abbildungsfunktionen unter Verwendung vorgegebener Logikgatter sehr aufwendig zu realisieren ist. Auch hinsichtlich der erreichbaren Schaltgeschwindigkeit sind die komplizierten Logikgatter verbesserungsbedürftig. Die Einschränkung des Umfangs der möglichen Logikfunktionen kompliziert die automatische Logikpartitionierung bei einem FPGA-Entwurf erheblich.

**Aufgabenstellung**

**[0009]** Der Erfindung liegt insbesondere das Problem zugrunde, eine Logik-Schaltkreis-Anordnung mit einer alternativen Architektur bereitzustellen.

**[0010]** Das Problem wird durch eine Logik-Schaltkreis-Anordnung mit den Merkmalen gemäß dem unabhängigen Patentanspruch gelöst.

**[0011]** Die erfindungsgemäße Logik-Schaltkreis-Anordnung enthält mindestens zwei Datensignal-Eingänge, an denen mindestens zwei Datensignale bereitstellbar sind. Ferner weist die Logik-Schaltkreis-Anordnung einen mit den Datensignal-Eingängen gekoppelten ersten Signalpfad mit einer Mehrzahl von Transistoren eines ersten Leitungstyps auf, wobei die Transistoren derart miteinander verschaltbar sind, dass sie eine erste Logikfunktion von allen möglichen Logikfunktionen zur Logik-Verknüpfung der zwei Datensignale realisieren, so dass ein das Ergebnis der ersten Logikfunktion repräsentierendes Ausgangssignal bereitgestellt wird. Ferner weist die Logik-Schaltkreis-Anordnung eine Mehrzahl von mit den Transistoren gekoppelten Steuereingängen auf, mittels welcher ein vorgebbarer, von den Transistoren gebildeter Signallaufpfad zu- oder abschaltbar ist.

**[0012]** Eine Grundidee der Erfindung besteht darin, eine universell konfigurierbare Logikzelle bereitzustellen, mittels welcher an den Datensignal-Eingängen bereitstellbare Datensignale miteinander gemäß einer vorgebaren Logikfunktion verknüpfbar sind. Diese Logikfunktion ist eine ganz bestimmte Logikfunktion, die aus allen denkbaren Logikfunktionen zur Verknüpfung der Datensignale auswählbar ist. Mit anderen Worten kann die realisierte Logikfunktion jede mögliche Logikverknüpfung der Datensignale sein. Mittels der fest oder variabel vorgebbaren Verschaltung der Transistoren in dem ersten Signalpfad ist die jeweils durchzuführende Logikfunktion festlegbar. Hierfür werden an den Steuereingängen der Logik-Schaltkreis-Anordnung entsprechende elektrische Signale angelegt. Diese Signale können zum Beispiel variable Signale sein, welche die Transistoren derart ansteuern, dass ein gewünschter, von den Transistoren gebildeter Signallaufpfad zu- oder abgeschaltet wird, wobei dem so ausgewählten Signallaufpfad (der anschaulich dem Laufweg der Datensignale durch die Anordnung miteinander verschalteter Transistoren entspricht) gerade die gewünschte Logikfunktion zugeordnet ist. Bei Anlegen variabler, auf eine bestimmte Logikfunktion abgestimmter Signale an den Steuereingängen wird anschaulich ein Sperren/Durchlassen der Transistoren eingestellt, wodurch nur bestimmte Signallaufpfade ermöglicht werden, und andere nicht. Alternativ können die Transistoren hardwaremäßig unveränderlich miteinander verschaltet sein, wodurch erlaubte Signallaufpfade und folglich eine bestimmte Logikfunktion fest vorgegeben sind. An den Steuereingängen können dann zum Beispiel ein elektrisches Versorgungsspannungs-Potential oder ein elektrisches Masse-Potential bereitgestellt sein. Anders ausgedrückt können zum Beispiel mittels Vorgebens von Logikfunktionssignalen an den Steuereingängen der Transistoren variable Logikfunktionssignale angelegt werden. Alternativ kann die Logik festverdrahtet bzw. unveränderlich realisiert sein, indem aufgrund der Verschaltung der Transistoren (z.B. mittels Kurzschließens einzelner Transistoren) Transistor-Steueranschlüssen fest vorgegebene elektrische Potentiale zugewiesen sind, in welchen eine fest zugewiesene Logikfunktion kodiert ist.

**[0013]** Bei der erfindungsgemäßen Architektur sind weder Look-Up-Tabellen noch Gatter-Abfolgen zur Realisierung der Logik-Grundfunktionen erforderlich. Mit dem erfindungsgemäßen Transistornetzwerk, das aus dem Signalpfad von Transistoren gebildet ist, kann mittels Vorgebens der Verschaltung der Transistoren jede beliebige aller möglichen Logikfunktionen für die jeweilige Zahl von Eingangssignalen realisiert werden.

**[0014]** Somit ist eine neue Architektur einer universell konfigurierbaren Logikzelle geschaffen, die sich insbesondere für den Einsatz in PLDs, weiter insbesondere für den Einsatz in FPGAs bzw. MPGAs eignet. Mit der erfindungsgemäßen Realisierung ist eine geringe erforderliche Chipfläche zum Realisieren der Logikoperationen mit einer sehr geringen Verlustleistung und einer hohen Schaltgeschwindigkeit kombiniert. Die erfindungsgemäße Logik-Schaltkreis-Anordnung weist eine hohe Störsicherheit auf und eignet sich zur flexiblen Skalierung für eine beliebige Anzahl von Eingängen und Ausgängen. Bei Verwendung von genau einem Signalpfad von Transistoren ist eine besonders platzsparende und schnelle Logik-Architektur erreicht.

**[0015]** Im Unterschied zu den aus dem Stand der Technik bekannten LUT-Architekturen ist es bei der Erfindung nicht zwingend erforderlich, die zum Auswählen einer Logikfunktion erforderlichen zum Beispiel 2<sup>n</sup> Funktionswerte in einer extern vorgesehenen Speicherzelle abzulegen. Gegenüber dem statischen LUT-Prinzip weist die Erfindung eine deutlich erhöhte Flexibilität auf, da sie das benutzerdefinierte bzw. variable Vorgeben einer Logikfunktion mittels Bereitstellens von Signalen an den Steuereingängen der Logik-Schaltkreis-Anordnung ermöglicht. Allerdings ist es erfindungsgemäß alternativ möglich, einer Logik-Schaltkreis-Anordnung in einem bestimmten Anwendungsszenario eine gewünschte Logikfunktion fest vorzugeben, beispielsweise mittels fest vorgegebenen Verschaltens der Transistoren oder mittels Anlegens fest vorgegebener elektrischer Potentiale an die unterschiedlichen Steueranschlüsse der Transistoren des Signalpfads, welche Transistoren mittels der Steuereingänge der Logik-Schaltkreis-Anordnung ansteuerbar sind. Das direkte Vorgeben einer gewünschten Logikfunktion mittels entsprechenden Steuerns der Transistoren der Signallaufpfade erhöht die Störsicherheit gegenüber der LUT-Lösung.

**[0016]** Bei einer Ausgestaltung der erfindungsgemäßen Logik-Schaltkreis-Anordnung mit einer variablen Logik, die mittels Vorgebens von Logikfunktionssignalen eingestellt wird, eignet sich die Logik-Schaltkreis-Anord-

nung als FPGA. In einer Implementierung der erfindungsgemäßen Logik-Schaltkreis-Anordnung als Structured ASIC (strukturierter anwenderprogrammierbarer integrierter Schaltkreis) bzw. als Mask Programmable Gate-Array (MPGA) wird die Logikfunktion der Logik-Schaltkreis-Anordnung dagegen fest eingestellt, beispielsweise mittels fest vorgegebenen Verschaltens unterschiedlicher Transistoren unter Verwendung von Vias. Dadurch ist ein Logikchip mit geringem Aufwand herstellbar, wobei nur eine sehr geringe Anzahl von teuren Masken erforderlich ist.

**[0017]** Die erfindungsgemäße Logik-Schaltkreis-Anordnung ermöglicht eine erwünschte Logikfunktion mit gegenüber der Realisierung mit Einzelgattern gemäß dem Stand der Technik deutlich verringertem schaltungs-technischen Aufwand.

**[0018]** Es ist anzumerken, dass erfindungsgemäß eine gewünschte Logikfunktion einzig und allein mittels Vorgebens von Signalen an den Steueranschlüssen ausgewählt werden kann, wohingegen die Datensignale, welche an den Datensignal-Eingängen bereitstellbar sind, auf die ausgewählte Logikfunktion vorzugsweise keinerlei Einfluss haben. Dies unterscheidet die erfindungsgemäße Lösung wesentlich von dem aus (3) bekannten FPGA, bei dem erst durch das Zusammenspiel aus Datensignalen und zusätzlichen anderen Signalen bestimmte Datenpfade in einem Multiplexer eingestellt werden, wodurch die gewünschte Logikfunktion ausgewählt wird.

**[0019]** Die erfindungsgemäße Logik-Schaltkreis-Anordnung weist mindesten zwei Datensignal-Eingänge auf. Für typische Anwendungssituationen werden häufig eine größere Zahl von Datensignal-Eingängen vorgesehen, zum Beispiel drei, vier, fünf oder mehr Datensignal-Eingänge.

**[0020]** Bevorzugte Weiterbildungen der Erfindung ergeben sich aus den abhängigen Ansprüchen.

**[0021]** Die Logik-Schaltkreis-Anordnung kann einen Evaluierungs-Schalter und einen Vorlade-Schalter aufweisen, welche Schalter derart verschaltet und steuerbar sind, dass an einem Ausgang der Logik-Schaltkreis-Anordnung bei geöffnetem Evaluierungs-Schalter (und simultan geschlossenem Vorlade-Schalter) das Ausgangssignal bereitgestellt ist, und dass an dem Ausgang der Logik-Schaltkreis-Anordnung bei geöffnetem Vorlade-Schalter (und simultan geschlossenem Evaluierungs-Schalter) ein Referenzsignal (zum Beispiel die Versorgungsspannung oder das Massepotential) bereitgestellt ist.

**[0022]** Gemäß dieser Ausgestaltung kann mittels des Vorlade- oder Precharge-Schalters der Ausgang auf das Referenzpotential vorgeladen werden (Precharge-Phase), zum Beispiel während der ersten Hälfte einer Schaltperiode der Logik-Schaltkreis-Anordnung. Mittels des Evaluierungs-Schalters kann an dem Ausgang das gemäß der vorgegebenen Logikfunktion verarbeitete Ausgangssignal bereitgestellt werden, zum Beispiel, während der zweiten Hälfte der Schaltperiode der Logik-Schaltkreis-Anordnung (Evaluate-Phase).

**[0023]** Der Evaluierungs-Schalter und der Vorlade-Schalter können jeweils Transistoren sein, zum Beispiel Feldeffekttransistoren oder Bipolartransistoren.

**[0024]** Gemäß einer anderen Ausgestaltung der Logik-Schaltkreis-Anordnung ist ein mit den Datensignal-Eingängen gekoppelter zweiter Signalpfad mit einer Mehrzahl von Transistoren eines zu dem ersten Leitungstyp komplementären zweiten Leitungstyps vorgesehen. Die Transistoren sind derart miteinander verschaltbar, dass sie eine zweite Logikfunktion von allen möglichen Logikfunktionen zur Logik-Verknüpfung der zwei Datensignale realisieren, so dass ein das Ergebnis der zweiten Logikfunktion repräsentierendes Ausgangssignal bereitgestellt wird. Das Ergebnis der zweiten Logikfunktion ist zu dem Ergebnis der ersten Logikfunktion invers.

**[0025]** Zum Beispiel können die Transistoren des ersten Leitungstyps als n-Kanal-Transistoren vorgesehen sein und die Transistoren des zweiten Leitungstyps als p-Kanal-Transistoren, oder umgekehrt.

**[0026]** Anschaulich ist somit eine CMOS-artige Realisierung der Logik-Schaltkreis-Anordnung geschaffen.

**[0027]** Mathematisch können die aus der boolschen Logik resultierenden Produktterme einer Logik-Funktion als Serienpfade von p-Kanal-Transistoren bzw. n-Kanal-Transistoren aufgebaut werden. Jeweils einander ausschließende Produktterme können unter Verwendung eines Schalters zusammengefasst werden. Die erfindungsgemäße Logik-Schaltkreis-Anordnung zeichnet sich durch eine hohe Störsicherheit aus. Unter Verwendung von 2<sup>n</sup> Schaltern, z.B. Transistoren, kann jede von

$2^{2^n}$ 

möglichen Logikfunktionen benutzerdefiniert bzw. anwendungsspezifisch eingestellt werden.

**[0028]** Zumindest ein Teil der ersten und/oder der zweiten Transistoren können als Feldeffekttransistoren oder Bipolartransistoren gebildet sein. Die Transistoren werden zum Beispiel in Siliziumtechnologie oder unter Verwendung von III-V-Halbleitern realisiert. Die Transistoren können in Bulk-Silizium- oder in SOI-Technologie ("Silicon-on-Insulator"), als Fin-FET-Transistoren oder als MOSFETs gebildet sein.

**[0029]** Der erste Leitungstyp der Transistoren kann der p-Leitungstyp und der zweite Leitungstyp der n-Leitungstyp sein. Alternativ kann der erste Leitungstyp der n-Leitungstyp und der zweite Leitungstyp der p-Leitungstyp sein. Allerdings sind die Leitungstypen der Transistoren in den beiden Signalpfaden jeweils invers bzw. komplementär zueinander.

**[0030]** Die Schaltkreis-Anordnung kann als CMOS-Schaltkreis-Anordnung eingerichtet sein.

**[0031]** Die Schaltkreis-Anordnung der Erfindung kann als Application-Specific Integrated Circuit (ASIC) vorgesehen sein. Insbesondere kann die Schaltkreis-Anordnung als Programmable Logical Device (PLD), als Field-Programmable Gate-Array (FPGA) oder als maskenprogrammierter Application-Specific Integrated Circuit vorgesehen sein.

**[0032]** Die Mehrzahl von Logikfunktionen, die mit der erfindungsgemäßen Logik-Schaltkreis-Anordnung realisiert werden können, ermöglichen vorzugsweise zumindest eine der folgenden Logikoperationen: UND-Verknüpfung, ODER-Verknüpfung, Nicht-UND-Verknüpfung, Nicht-ODER-Verknüpfung, Exklusiv-ODER-Verknüpfung, Exklusiv-Nicht-ODER-Verknüpfung. Vorzugsweise ist die erfindungsgemäße Logik-Schaltkreis-Anordnung derart eingerichtet, dass mittels zugehöriger Konfigurierung jede hinsichtlich der Verknüpfung einer vorgegebenen Anzahl von Eingabesignalen mögliche Logikoperation realisierbar ist.

**[0033]** Bei zumindest einem Teil der Transistoren kann mindestens ein Logikfunktionssignal zum Vorgeben der ersten bzw. zweiten Logikfunktion unveränderlich bzw. unveränderbar vorgegeben werden. Bei dieser Alternative ist anschaulich die von der Logik-Schaltkreis-Anordnung durchführbare Logikoperation fest voreingestellt, zum Beispiel mittels Bereitstellens fester elektrischer Potentiale an Steuereingänge von Transistoren der Signalpfade oder mittels entsprechenden Verschaltens der Transistoren unter Verwendung zugeordneter elektrischer Kontaktierungselemente. Anders ausgedrückt kann das feste Voreinstellen hardwaremäßig realisiert sein.

**[0034]** Bei der Ausgestaltung der Logik-Schaltkreis-Anordnung mit unveränderlich vorgegebener Logikfunktion bzw. Logikoperation kann zumindest ein Teil der Transistoren mit einer Speicher-Einrichtung gekoppelt sein, in welcher die Werte des mindestens einen Logikfunktionssignals gespeichert sind. In einer solchen Speicher-Einrichtung sind die Werte der an die Steuereingänge der Transistoren anzulegenden elektrischen Potentiale (vorzugsweise digital, das heißt mittels Logik-Werte "1" oder "0") kodiert. Daher ist anschaulich in der in der Speicher-Einrichtung enthaltenen Information die Logikfunktion festgelegt, welche von der Logik-Schaltkreis-Anordnung realisiert wird.

**[0035]** Alternativ kann zumindest einem Teil der Transistoren mindestens ein Logikfunktionssignal zum Vorgeben der ersten und zweiten Logikfunktion variabel vorgegeben sein. Bei dieser Ausgestaltung der Logik-Schaltkreis-Anordnung kann diese jede beliebige Logikfunktion durchführen, welche der Schaltkreis-Anordnung mittels eines (zum Beispiel zeitlich) veränderlichen Potentials an den Steuereingängen der Transistoren vorgegeben ist. In diesem Szenario kann die Logik-Schaltkreis-Anordnung als variable Logikzelle in einem übergeordneten Schaltkreis verschaltet werden, was flexible Schaltungsarchitekturen ermöglicht. Eine Logikzelle der Erfindung kann anders ausgedrückt in einen komplexen Schaltkreis eingebettet sein und in diesem im Rahmen einer Gesamtfunktionalität eine Logik-Teilaufgabe erfüllen, wobei die von der Logikzelle erfüllte Logik-Teilaufgabe bedarfsweise mittels Vorgebens von Potentialen an Transistoren geändert bzw. neu eingestellt werden kann.

**[0036]** Die Datensignale, Ausgangssignale und Logikfunktionssignale sind vorzugsweise digitale Signale, von denen jedes einen Logik-Wert "1" oder "0" annehmen kann.

**[0037]** Zwischen den mindestens zwei Datensignal-Eingängen einerseits und den ersten Signalpfaden und den zweiten Signalpfaden andererseits kann eine Einrichtung zum Generieren des zu einem jeweiligen Daten-

signal logisch komplementären Datensignals angeordnet sein. Gemäß dieser Ausgestaltung kann ein zu einem Datensignal logisch inverses bzw. komplementäres Datensignal generiert werden und den ersten bzw. zweiten Signalpfaden bereitgestellt werden. Eine solche Einrichtung kann für jedes der Datensignale einen zugeordneten Inverter-Teilschaltkreis aufweisen, welcher beispielsweise mittels eines p-MOS-Transistors und eines n-MOS-Transistors realisiert sein kann.

[0038] Den ersten Signalpfaden und den zweiten Signalpfaden kann eine Ausgangssignalweiterverarbeitungs-Einrichtung nachgeschaltet sein. Diese kann zum Weiterverarbeiten des Ausgangssignals dienen, welches die Logik-Schaltkreis-Anordnung entsprechend der zugeordneten Logikfunktionen generiert.

[0039] Diese Ausgangssignalweiterverarbeitungs-Einrichtung kann für den Fall von mindestens zwei Datensignal-Eingängen einen Inverter aufweisen. Dieser Inverter oder Inverter-Schaltkreis kann zum Bilden des logisch Inversen zu einem von dem Logik-Schaltkreis bereitgestellten Ausgangssignals eingerichtet sein, z.B. in einem Szenario, in dem ein solches logisch Inverses für nachgeschaltete Komponenten bereitgestellt werden soll. Im Fall von drei oder mehr Datensignal-Eingängen kann die Ausgangssignalweiterverarbeitungs-Einrichtung zusätzlich oder alternativ zu dem mindestens einen Inverter eine Multiplexer-Einrichtung aufweisen, welche eines oder mehrere von der Logik-Schaltkreis-Anordnung generierte Signale weiterverarbeitet.

[0040] Die Logik-Schaltkreis-Anordnung ist vorzugsweise zum Verarbeiten von digitalen Datensignalen und digitalen Logikfunktionssignalen eingerichtet. Jedoch kann die erfindungsgemäße Schaltkreis-Anordnung alternativ auch zum Verarbeitung analoger Signale eingerichtet sein.

[0041] Die Transistoren der Logik-Schaltkreis-Anordnung können zum Verarbeiten von zwei Datensignalen  $a_0, a_1$  zu einem Ausgangssignal  $y(a_0, a_1)$  unter Verwendung der Logikfunktionssignale  $c_0, c_1, c_2, c_3$  zum Vorgeben der Logikfunktionen gemäß folgender Abbildungsvorschrift verschaltet sein:

$$y = \overline{c_0} \cdot \overline{a_1} \cdot \overline{a_0} \vee \overline{c_1} \cdot \overline{a_1} \cdot a_0 \vee \overline{c_2} \cdot a_1 \cdot \overline{a_0} \vee \overline{c_3} \cdot a_1 \cdot a_0 \quad (1)$$

und

$$\overline{y} = c_0 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_1 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_2 \cdot \overline{a_1} \cdot a_0 \vee c_3 \cdot \overline{a_1} \cdot a_0 \quad (2)$$

[0042] Die Datensignale, das Ausgangssignal und die Logikfunktionssignale sind vorzugsweise digitale Signale, von denen jedes einen logischen Wert "1" oder "0" annehmen kann.

[0043] Ein Querstrich über einer jeweiligen Variable oder Konstante oder einer Verknüpfung davon bedeutet "das logisch Komplementäre" der Variablen oder Konstanten oder Verknüpfung.

[0044] Alternativ können die Transistoren der Logik-Schaltkreis-Anordnung zum Verarbeiten von drei Datensignalen  $a_0, a_1, a_2$  zu einem Ausgangssignal  $y(a_0, a_1, a_2)$  unter Verwendung der Logikfunktionssignale  $c_0, c_1, c_2, c_3, c_4, c_5, c_6, c_7$  zum Vorgeben der Logikfunktionen gemäß folgender Abbildungsvorschrift verschaltet sein:

$$y(a_2, a_1, a_0) = \overline{a_2} \cdot y_0(a_1, a_0) \vee a_2 \cdot y_1(a_1, a_0) \quad (3)$$

mit

$$y_0 = \overline{c_0} \cdot \overline{a_1} \cdot \overline{a_0} \vee \overline{c_1} \cdot \overline{a_1} \cdot a_0 \vee \overline{c_2} \cdot a_1 \cdot \overline{a_0} \vee \overline{c_3} \cdot a_1 \cdot a_0 \quad (4)$$

und

$$\overline{y_0} = c_0 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_1 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_2 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_3 \cdot \overline{a_1} \cdot \overline{a_0} \quad (5)$$

und

$$y_1 = \overline{c_4} \cdot \overline{a_1} \cdot \overline{a_0} \vee \overline{c_5} \cdot \overline{a_1} \cdot \overline{a_0} \vee \overline{c_6} \cdot \overline{a_1} \cdot \overline{a_0} \vee \overline{c_7} \cdot \overline{a_1} \cdot \overline{a_0} \quad (6)$$

und

$$\overline{y_1} = c_4 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_5 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_6 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_7 \cdot \overline{a_1} \cdot \overline{a_0} \quad (7)$$

[0045] Gemäß einer anderen Alternative können die Transistoren zum Verarbeiten von n Datensignalen  $a_0, a_1, \dots, a_{n-1}$  zu einem Ausgangssignal  $y(a_0, a_1, \dots, a_{n-1})$  unter Verwendung von  $2^n$  Logikfunktionssignalen  $c_0, c_1, \dots, c_{4(n-1)-1}$  zum Vorgeben der Logikfunktionen gemäß folgender Abbildungsvorschrift verschaltet sein:

$$y(a_{n-1}, \dots, a_1, a_0) = \overline{a_{n-1}} \dots \overline{a_2} \cdot \overline{y_0} \vee \overline{a_{n-1}} \dots \overline{a_2} \cdot \overline{y_1} \vee \dots \vee \overline{a_{n-1}} \dots \overline{a_2} \cdot \overline{y_{2^{(n-2)}-1}} \quad (8)$$

mit

$$y_0 = \overline{c_0} \cdot \overline{a_1} \cdot \overline{a_0} \vee \overline{c_1} \cdot \overline{a_1} \cdot \overline{a_0} \vee \overline{c_2} \cdot \overline{a_1} \cdot \overline{a_0} \vee \overline{c_3} \cdot \overline{a_1} \cdot \overline{a_0} \quad (9)$$

und

$$\overline{y_0} = c_0 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_1 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_2 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_3 \cdot \overline{a_1} \cdot \overline{a_0} \quad (10)$$

und

$$y_1 = \overline{c_4} \cdot \overline{a_1} \cdot \overline{a_0} \vee \overline{c_5} \cdot \overline{a_1} \cdot \overline{a_0} \vee \overline{c_6} \cdot \overline{a_1} \cdot \overline{a_0} \vee \overline{c_7} \cdot \overline{a_1} \cdot \overline{a_0} \quad (11)$$

und



$$\overline{y_1} = c_4 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_5 \cdot \overline{a_1} \cdot a_0 \vee c_6 \cdot a_1 \cdot \overline{a_0} \vee c_7 \cdot a_1 \cdot a_0 \quad (12)$$

...

und

$$\overline{y_2^{(n-2)-1}} = \overline{c_2^{n-4}} \cdot \overline{a_1} \cdot \overline{a_0} \vee \overline{c_2^{n-3}} \cdot \overline{a_1} \cdot a_0 \vee \overline{c_2^{n-2}} \cdot a_1 \cdot \overline{a_0} \vee \overline{c_2^{n-1}} \cdot a_1 \cdot a_0 \quad (13)$$

und

$$\overline{y_2^{(n-2)-1}} = c_2^{n-4} \cdot \overline{a_1} \cdot \overline{a_0} \vee c_2^{n-3} \cdot \overline{a_1} \cdot a_0 \vee c_2^{n-2} \cdot a_1 \cdot \overline{a_0} \vee c_2^{n-1} \cdot a_1 \cdot a_0 \quad (14)$$

[0046] Gemäß einer anderen Ausgestaltung der Logik-Schaltkreis-Anordnung können die Transistoren verschaltet sein zum Verarbeiten von n Datensignalen  $a_0, a_1, \dots, a_{n-1}$  zu einem Ausgangssignal  $y(a_0, a_1, \dots, a_{n-1})$  unter Verwendung von Logikfunktionssignalen  $c_0, c_1, \dots, c_{2^n(n-1)}$  mit  $c_i \in \{1, 0\}$  zum Vorgeben der Logikfunktion gemäß folgender Abbildungsvorschrift:

$$y = c_0 \cdot \overline{a_{n-1}} \cdot \dots \cdot \overline{a_1} \cdot \overline{a_0} \vee \dots \vee c_{2^n(n-1)} \cdot a_{n-1} \cdot \dots \cdot a_1 \cdot a_0 \vee c_{2^n(n-1)} \cdot a_{n-1} \cdot \dots \cdot a_1 \cdot a_0 \quad (15)$$

[0047] Anschaulich stellt eine auf Transistor-Basis realisierte Logik-Schaltkreis-Anordnung unter Verwendung des aus Gleichung (15) ersichtlichen Prinzips eine direkte, von einer LUT-basierten-Verarbeitung freie Realisierung aller Funktionen von n Eingängen bzw. Datensignalen dar. Hierbei treten (analog wie bei dem Fall einer Zelle mit zwei Eingängen)  $(n+1)$  lange Serienpfade (n Eingänge und ein Konfigurationstransistor) von p-MOS- bzw. n-MOS-Transistoren auf.

[0048] Zum Darstellen von allen  $2^n$  Produkttermen werden  $2^n$  Serienpfade im Pull-Up-Pfad und im Pull-Down-Pfad gebildet, und durch einen gemeinsamen Ausgang  $y_{im}$  gekoppelt.

[0049] Zum Beispiel ergeben sich für den Fall von vier Eingängen fünf Transistoren pro Serienpfad, wobei für den Pull-Up-Pfad (z.B. erster Signalpfad) und für den Pull-Down-Pfad (z.B. zweiter Signalpfad) jeweils sechzehn Serienpfade erforderlich sind. Somit sind – ohne Berücksichtigung von Transistoren für Eingangs- bzw. Ausgangsinverter –  $5 \cdot 2 \cdot 16 = 160$  Transistoren erforderlich, um mit der beschriebenen Ausgestaltung eine Logik-Funktion aus allen Logikfunktionen zu realisieren, die vier Datensignale miteinander verknüpfen.

[0050] Die Transistoren der Signalpfade können derart miteinander verschaltet sein, dass einzig mittels Vorgebens des mindestens einen Logikfunktionssignals eine zugeordnete Logikfunktion eindeutig vorgegeben ist. Mit anderen Worten ist die gewählte Logikfunktion vorzugsweise durch die Werte der Potentiale an Steuereingängen der Transistoren eindeutig bestimmt, insbesondere ist die ausgewählte Logikfunktion vorzugsweise unabhängig von den Werten der Datensignale.

#### Ausführungsbeispiel

[0051] Ausführungsbeispiele der Erfindung sind in den Figuren dargestellt und werden im Weiteren näher erläutert.

[0052] Es zeigen:

[0053] Fig. 1 eine Logik-Schaltkreis-Anordnung gemäß seinem ersten Ausführungsbeispiel der Erfindung,

[0054] Fig. 2 eine Tabelle, in der eine Zuordnung zwischen Logikfunktionssignalen und zugeordneten Logikfunktionen gemäß einem Ausführungsbeispiel der Erfindung mit zwei Datensignalen dargestellt sind,

[0055] Fig. 3 eine Logik-Schaltkreis-Anordnung gemäß einem zweiten Ausführungsbeispiel der Erfindung,

[0056] Fig. 4 eine Logik-Schaltkreis-Anordnung gemäß einem dritten Ausführungsbeispiel der Erfindung,

[0057] Fig. 5 eine schematische Layout-Anordnung von Leiterbahnen gemäß einer Verschaltung der Transistoren der erfindungsgemäßen Logik-Schaltkreis-Anordnung gemäß einem Ausführungsbeispiel der Erfindung,

[0058] Fig. 6 eine Logik-Schaltkreis-Anordnung gemäß einem vierten Ausführungsbeispiel der Erfindung,

[0059] Fig. 7 eine Logik-Schaltkreis-Anordnung gemäß einem fünften Ausführungsbeispiel der Erfindung.

[0060] Gleiche oder ähnliche Komponenten in unterschiedlichen Figuren sind mit gleichen Bezugsziffern versehen.

[0061] Die Darstellungen in den Figuren sind schematisch und nicht maßstäblich.

[0062] Im Weiteren wird bezugnehmend auf Fig. 1 eine Logik-Schaltkreis-Anordnung **100** gemäß einem ersten Ausführungsbeispiel der Erfindung beschrieben.

[0063] Die Logik-Schaltkreis-Anordnung **100** weist einen ersten Datensignal-Eingang **101** und einen zweiten Datensignal-Eingang **102** auf, an denen zwei Datensignale  $a_0$  bzw.  $a_1$  bereitgestellt sind. Die Logik-Schaltkreis-Anordnung **100** enthält eine mit den Datensignal-Eingängen **101**, **102** gekoppelte erste Signalpfad-Einheit **103** mit einer Mehrzahl von n-MOS-Transistoren **104** (n-Leitungstyp), wobei die n-MOS-Transistoren **104** derart miteinander verschaltet sind, dass sie eine erste Logikfunktion von einer Mehrzahl von Logikfunktionen zur Logik-Verknüpfung der zwei Datensignale  $a_0$ ,  $a_1$  realisieren, so dass ein das Ergebnis der ersten Logikfunktion repräsentierendes Ausgangssignal an Ausgang **107** bereitgestellt wird. Ferner enthält die Logik-Schaltkreis-Anordnung **100** eine mit den Datensignal-Eingängen **101**, **102** gekoppelte zweite Signalpfad-Einheit **105** mit einer Mehrzahl von p-MOS-Transistoren **106** (des p-Leitungstyps, welcher zu dem n-Leitungstyps komplementär ist). Die p-MOS-Transistoren **106** sind miteinander derart verschaltet, dass sie eine zweite Logikfunktion von einer Mehrzahl unterschiedlicher Logikfunktionen zur Logik-Verknüpfung der zwei Datensignale  $a_0$ ,  $a_1$  realisieren, so dass ein das Ergebnis der zweiten Logikfunktion repräsentierendes Ausgangssignal an Ausgang **107** bereitgestellt wird, wobei das Ergebnis der zweiten Logikfunktion invers zu dem Ergebnis der ersten Logikfunktion ist. Zwischen Ausgang **107** und einem globalen Ausgang **109** ist eine Weiterverarbeitungseinheit **108** geschaltet, mittels welcher Ausgangssignale weiterverarbeitet werden können, zum Bereitstellen eines weiterverarbeiteten Ausgangssignals  $y$  an dem globalen Ausgang **109**. An dem globalen Signalausgang **109** der Logik-Schaltkreis-Anordnung **100** ist das Ausgangssignal  $y$  bereitgestellt, welches die Logik-Verknüpfung der Eingabesignale  $a_0$ ,  $a_1$  entsprechend der ausgewählten Logik darstellt und bereits einer Weiterverarbeitung unterzogen worden ist.

[0064] Wie ferner in Fig. 1 gezeigt, ist den Signalpfad-Einheiten **103**, **105** an Steuereingängen jeweils ein erstes Logikfunktionssignal  $c_0$  und ein zweites Logikfunktionssignal  $c_1$  bereitgestellt. Mittels Vorgebens dieser Logikfunktionssignale  $c_0$ ,  $c_1$  werden die Transistoren **104** bzw. **106** der Signalpfad-Einheit **103** bzw. **105** derart angesteuert, dass die Signale  $a_0$ ,  $a_1$  von den Signalpfad-Einheiten **103** bzw. **105** entsprechend der ersten Logikfunktion bzw. der zweiten Logikfunktion miteinander verknüpft werden. Somit wird anschaulich mittels Vorgebens der logischen Werte der Logikfunktionssignale  $c_0$ ,  $c_1$  eine ganz bestimmte Logikfunktion ausgewählt.

[0065] Die Logikfunktionssignale  $c_i$ , die Datensignale  $a_i$  und das Ausgangssignal  $y$  können jeweils entweder einen logischen wert "1" oder einen logischen Wert "0" annehmen.

[0066] Im Weiteren wird die der erfindungsgemäßen Lösung zugrundeliegende theoretische Basis, die auf der boolschen Logik beruht, beschrieben.

[0067] Eine boolsche Funktion lässt sich in der kanonisch-konjunktiven Normalform als ODER-Verknüpfung

der Produktterme ihrer  $n$  Eingänge ausdrücken (in **Fig. 1** beispielsweise ist  $n=2$ , da zwei Eingangssignale  $a_1, a_0$  bereitgestellt sind). Diesen  $n$  Eingängen sind  $2^n$  Produktterme zugeordnet.

**[0068]** Auf Standard-CMOS-Logik angewendet werden erfindungsgemäß die Produktterme für den logischen Wert "1" einer Funktion als Serienpfad von p-Kanal-Transistoren realisiert (in **Fig. 1** beispielsweise: p-Kanal-Transistoren **106**). Der Logik-Wert "0" wird entsprechend als Serienpfad aus n-Kanal-Transistoren realisiert (in **Fig. 1**: n-MOS-Transistoren **104**). Entsprechend kann jede Logik-Funktion, gemäß welcher an  $n$  Eingängen bereitgestellte Signale miteinander Logik-verknüpft werden, aus  $2^n$  Produkttermen zusammengesetzt werden, indem anschaulich Produktterme zu- oder abgeschaltet werden.

**[0069]** Für zwei Eingänge  $a_0$  und  $a_1$  (wie in **Fig. 1**) gilt:

$$y = k_0 \cdot \overline{a_1} \cdot \overline{a_0} \vee k_1 \cdot \overline{a_1} \cdot a_0 \vee k_2 \cdot a_1 \cdot \overline{a_0} \vee k_3 \cdot a_1 \cdot a_0 \quad (16)$$

$$\overline{y} = k_4 \cdot \overline{a_1} \cdot a_0 \vee k_5 \cdot a_1 \cdot \overline{a_0} \vee k_6 \cdot \overline{a_1} \cdot \overline{a_0} \vee k_7 \cdot a_1 \cdot a_0 \quad (17)$$

mit

$$k_i = \{0, 1\} \text{ mit } i=0, 1, \dots, 7 \quad (18)$$

**[0070]** Jede Funktion  $y=f(a_0, a_1)$  wird gebildet, indem vier Werte der Schaltkoeffizienten oder Logikfunktionsvariablen  $k_0$  bis  $k_7$  auf einen Wert logisch "1" und die restlichen auf einen Wert logisch "0" gesetzt werden. Da in CMOS-Logik die p-Kanal-Transistoren mit einem elektrischen Potential "0" am Steuer- oder Gate-Anschluss öffnen, die n-Kanal-Transistoren hingegen bei einem elektrischen Potential mit einem Wert "1", lassen sich die Produktterme in Gleichungen (16), (17) zu einander sich ausschließenden Paaren ordnen. In Gleichungen (16), (17) schließen sich die jeweils ersten Produktterme gegenseitig logisch aus, die jeweils zweiten, die jeweils dritten und die jeweils vierten.

**[0071]** Für die Schaltkoeffizienten  $k_i$  gilt die Beziehung:

$$k_0 = \overline{k_4}, \quad k_1 = \overline{k_5}, \quad k_2 = \overline{k_6}, \quad k_3 = \overline{k_7} \quad (19)$$

**[0072]** Aus Gleichungen (16), (17), (19) ergibt sich nach Zusammenfassen zu vier unabhängigen Schaltvariablen  $c_0, c_1, c_2, c_3$ :

$$y = c_0 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_1 \cdot \overline{a_1} \cdot a_0 \vee c_2 \cdot a_1 \cdot \overline{a_0} \vee c_3 \cdot a_1 \cdot a_0 \quad (20)$$

$$\overline{y} = c_0 \cdot \overline{a_1} \cdot a_0 \vee c_1 \cdot a_1 \cdot \overline{a_0} \vee c_2 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_3 \cdot a_1 \cdot a_0 \quad (21)$$

mit

$$c_i = \{0, 1\} \text{ mit } i=0, 1, \dots, 3 \quad (22)$$

**[0073]** Anschaulich entspricht Gleichung (20) den Pfad von p-MOS-Transistoren **106** in **Fig. 1**, wohingegen Gleichung (21) dem Pfad von n-MOS-Transistoren **104** in **Fig. 1** entspricht.

**[0074]** Die Logikfunktion zum Bilden von Ausgangssignal  $y$  wird konfiguriert, wie es in der Tabelle 200 aus **Fig. 2** gezeigt ist. In Tabelle 200 ist gezeigt, mittels welcher Koeffizienten  $c_i$  welche zugehörige Logikfunktion  $y$  eingestellt werden kann bzw. voreingestellt ist.

**[0075]** Die erfindungsgemäße Lösung zeichnet sich durch eine erhöhte Störsicherheit der Schaltung und durch eine sehr kompakte Realisierbarkeit (Layout der Logikzelle) aus. Ferner ist die Lösung sehr gut in Bezug auf die Anzahl von Ein- und Ausgängen skalierbar.

**[0076]** In bestimmten Anwendungsszenarien weist die universelle Logikzelle unter vergleichbaren Randbe-

dingungen im Vergleich mit konkurrierenden Lösungen eine verringerte Schaltverzögerung bei einer verringerten Energieaufnahme auf.

**[0077]** Die Erfindung kann im Rahmen der FPGA-Technologie oder als kombinatorische Kernzelle eines Structured ASICs eingesetzt werden. Darüber hinaus kann die Erfindung überall dort besonders vorteilhaft verwendet werden, wo die Implementierung einer Untermenge Logik-Funktionen aufwendiger ist als die Verwendung einer universellen Logikzelle.

**[0078]** Im Weiteren wird bezugnehmend auf **Fig. 3** eine halbleitertechnologische Realisierung der Funktion gemäß Gleichungen (20), (21) auf Basis von Transistoren beschrieben.

**[0079]** Bei der Logik-Schaltkreis-Anordnung **300** aus **Fig. 3** ist an einem ersten Datensignal-Eingang **101** ein erstes Datensignal  $a_0$  bereitgestellt. Ferner wird an einem zweiten Datensignal-Eingang **102** ein zweites Datensignal  $a_1$  bereitgestellt. Mittels eines ersten Inverter-Schaltkreises **301** wird aus dem ersten Datensignal  $a_0$  das dazu komplementäre Signal  $\overline{a_0}$  gebildet. Der erste Datensignal-Eingang **101** ist mit dem Gate-Bereich eines ersten n-MOS-Inverter-Transistors **302** gekoppelt. Ferner ist der erste Datensignal-Eingang **101** mit dem Gate-Anschluss eines ersten p-MOS-Inverter-Transistors **303** gekoppelt. Ein erster Source-/Drain-Bereich des ersten p-MOS-Inverter-Transistors **303** ist auf das elektrische Versorgungspotential **307** gebracht. Der zweite Source-/Drain-Bereich des ersten p-MOS-Inverter-Transistors **303** ist mit einem ersten Source-/Drain-Bereich des ersten n-MOS-Inverter-Transistors **302** gekoppelt, dessen zweiter Source-/Drain-Bereich auf das elektrische Massepotential **308** gebracht ist.

**[0080]** Ferner ist ein zweiter Inverter-Schaltkreis **304** vorgesehen, mittels welchen aus dem zweiten Datensignal  $a_1$  dessen logisch komplementäres Signal  $\overline{a_1}$  gebildet wird. Der zweite Datensignal-Eingang **102** ist mit den Gate-Anschlüssen eines zweiten n-MOS-Inverter-Transistors **305** und eines zweiten p-MOS-Inverter-Transistors **306** gekoppelt, welche Transistoren **305**, **306** den zweiten Inverter-Schaltkreis **304** bilden. Ein erster Source-/Drain-Bereich des zweiten p-MOS-Inverter-Transistors **306** ist mit dem ersten Source-/Drain-Bereich des ersten p-MOS-Inverter-Transistors **303** gekoppelt, wohingegen der zweite Source-/Drain-Anschluss des zweiten p-MOS-Inverter-Transistors **306** mit einem ersten Source-/Drain-Anschluss des zweiten n-MOS-Inverter-Transistors **305** gekoppelt ist. Der zweite Source-/Drain-Anschluss des zweiten n-MOS-Inverter-Transistors **305** ist auf das elektrische Massepotential **308** gebracht.

**[0081]** Wie in **Fig. 3** gezeigt, werden die Datensignale und deren logisch komplementäre Werte einer Signalpfad-Einheit **309** bereitgestellt. An einem ersten Signalpfad-Eingang **310** ist das Signal  $\overline{a_0}$  bereitgestellt. An einem zweiten Signalpfad-Eingang **311** ist das Signal  $\overline{a_1}$  bereitgestellt. An einem dritten Signalpfad-Eingang **312** ist das Signal  $a_0$  bereitgestellt. An einem vierten Signalpfad-Eingang **313** ist das Signal  $a_1$  bereitgestellt.

**[0082]** Die Signalpfad-Einheit **309** ist gebildet aus ersten bis zwölften p-MOS-Logik-Transistoren **314** bis **325** und aus ersten bis zwölften n-MOS-Logik-Transistoren **326** bis **337**. Die ersten bis zwölften p-MOS-Logik-Transistoren **314** bis **325** bilden eine erste Signalpfad-Teileinheit, wohingegen die ersten bis zwölften n-MOS-Logik-Transistoren **326** bis **337** eine zweite Signalpfad-Teileinheit bilden.

**[0083]** An einem ersten Logikfunktions-Eingang **338** ist ein erstes Logikfunktionssignal  $c_0$  angelegt. An einem zweiten Logikfunktions-Eingang **339** ist ein zweites Logikfunktionssignal  $c_1$  bereitgestellt. An einem dritten Logikfunktions-Eingang **340** ist ein drittes Logikfunktionssignal  $c_2$  bereitgestellt. An einem vierten Logikfunktions-Eingang **341** ist ein viertes Logikfunktionssignal  $c_3$  bereitgestellt. Die Logikfunktions-Eingänge **338** bis **341** können auch als Steuereingänge der Logik-Schaltkreis-Anordnung **300** bezeichnet werden.

**[0084]** Der vierte Logikfunktions-Eingang **341** ist mit dem Gate-Anschluss des neunten p-MOS-Logik-Transistors **322** und mit dem Gate-Anschluss des ersten n-MOS-Logik-Transistors **326** gekoppelt. Der dritte Logikfunktions-Eingang **340** ist mit den Gate-Anschlüssen des zehnten p-MOS-Logik-Transistors **323** und des zweiten n-MOS-Logik-Transistors **327** gekoppelt. Der zweite Logikfunktions-Eingang **339** ist mit den Gate-Anschlüssen des elften p-MOS-Logik-Transistors **324** und des dritten n-MOS-Logik-Transistors **328** gekoppelt. Der erste Logikfunktions-Eingang **338** ist mit den Gate-Anschlüssen des zwölften p-MOS-Logik-Transistors **325** und des vierten n-MOS-Logik-Transistors **329** gekoppelt.

**[0085]** Der erste Datensignal-Eingang **310** ist mit dem Gate-Anschluss des fünften n-MOS-Logik-Transistors **330**, des sechsten p-MOS-Logik-Transistors **319**, des siebten n-MOS-Logik-Transistors **332** und des vierten p-MOS-Logik-Transistors **317** gekoppelt. Der zweite Datensignal-Eingang **311** ist mit den Gate-Anschlüssen des neunten n-MOS-Logik-Transistors **334**, des zehnten n-MOS-Logik-Transistors **335**, des dritten p-MOS-Lo-

gik-Transistors 316 und des achten p-MOS-Logik-Transistors 321 gekoppelt. Der dritte Datensignal-Eingang 312 ist mit den Gate-Anschlüssen des fünften p-MOS-Logik-Transistors 318, des sechsten n-MOS-Logik-Transistors 331, des siebten p-MOS-Logik-Transistors 320 und des achten n-MOS-Logik-Transistors 333 gekoppelt. Der vierte Datensignal-Eingang 313 ist mit den Gate-Anschlüssen des ersten p-MOS-Logik-Transistors 314, des zweiten p-MOS-Logik-Transistors 315, des elften n-MOS-Logik-Transistors 336 und des zwölften n-MOS-Logik-Transistors 337 gekoppelt.

[0086] Erste Source-/Drain-Anschlüsse der ersten bis vierten p-MOS-Logik-Transistoren 314 bis 317 sind auf das elektrische Potential der Versorgungsspannung 307 gebracht. Der zweite Source-/Drain-Anschluss des ersten p-MOS-Logik-Transistors 314 ist mit einem ersten Source-/Drain-Anschluss des fünften p-MOS-Logik-Transistors 318 gekoppelt, dessen zweiter Source-/Drain-Anschluss mit einem ersten Source-/Drain-Anschluss des neunten p-MOS-Logik-Transistors 322 gekoppelt ist. Der zweite Source-/Drain-Anschluss des zweiten p-MOS-Logik-Transistors 315 ist mit einem ersten Source-/Drain-Anschluss des sechsten p-MOS-Logik-Transistors 319 gekoppelt, dessen zweiter Source-/Drain-Anschluss mit einem ersten Source-/Drain-Anschluss des zehnten p-MOS-Logik-Transistors 323 gekoppelt ist. Der zweite Source-/Drain-Anschluss des dritten p-MOS-Logik-Transistors 316 ist mit einem ersten Source-/Drain-Anschluss des siebten p-MOS-Logik-Transistors 320 gekoppelt, dessen zweiter Source-/Drain-Anschluss mit einem ersten Source-/Drain-Anschluss des elften p-MOS-Logik-Transistors 324 gekoppelt ist. Der zweite Source-/Drain-Anschluss des vierten p-MOS-Logik-Transistors 317 ist mit einem ersten Source-/Drain-Anschluss des achten p-MOS-Logik-Transistors 321 gekoppelt, dessen zweiter Source-/Drain-Anschluss mit einem ersten Source-/Drain-Anschluss des zwölften p-MOS-Logik-Transistors 325 gekoppelt ist.

[0087] Die zweiten Source-/Drain-Anschlüsse der neunten bis zwölften p-MOS-Logik-Transistoren 322 bis 325 sind mit dem Ausgang 107 und mit ersten Source-/Drain-Anschlüssen der ersten bis vierten n-MOS-Logik-Transistoren 326 bis 329 gekoppelt. Der zweite Source-/Drain-Anschluss des ersten n-MOS-Logik-Transistors 326 ist mit einem ersten Source-/Drain-Anschluss des fünften n-MOS-Logik-Transistors 330 gekoppelt, dessen zweiter Source-/Drain-Anschluss mit einem ersten Source-/Drain-Anschluss des neunten n-MOS-Logik-Transistors 334 gekoppelt ist. Der zweite Source-/Drain-Anschluss des zweiten n-MOS-Logik-Transistors 327 ist mit einem ersten Source-/Drain-Anschluss des sechsten n-MOS-Logik-Transistors 331 gekoppelt, dessen zweiter Source-/Drain-Anschluss mit einem ersten Source-/Drain-Anschluss des zehnten n-MOS-Logik-Transistors 335 gekoppelt ist. Der zweite Source-/Drain-Anschluss des dritten n-MOS-Logik-Transistors 328 ist mit einem ersten Source-/Drain-Anschluss des siebten n-MOS-Logik-Transistors 332 gekoppelt, dessen zweiter Source-/Drain-Anschluss mit einem ersten Source-/Drain-Anschluss des elften n-MOS-Logik-Transistors 336 gekoppelt ist. Ferner ist der zweite Source-/Drain-Anschluss des vierten n-MOS-Logik-Transistors 329 mit einem ersten Source-/Drain-Anschluss des achten n-MOS-Logik-Transistors 333 gekoppelt, dessen zweiter Source-/Drain-Anschluss mit einem ersten Source-/Drain-Anschluss des zwölften n-MOS-Logik-Transistors 337 gekoppelt ist. Die zweiten Source-/Drain-Anschlüsse der neunten bis zwölften n-MOS-Logik-Transistoren 334 bis 337 sind miteinander gekoppelt und auf das elektrische Massepotential 308 gebracht.

[0088] An dem Ausgang 107 ist das Ausgangssignal  $y_{\text{int}}$  bereitgestellt.

[0089] Aus dem Ausgangssignal  $y_{\text{int}}$  an dem Ausgang 107 wird unter Verwendung eines dritten Inverter-Schaltkreises 342 das Logik-Inverse  $y_0$  gebildet, das an einem globalen Ausgang 345 bereitgestellt ist. Das Ausgangssignal  $y_{\text{int}}$  wird durch den dritten Inverter-Schaltkreis 342 geführt, gebildet aus einem dritten n-MOS-Inverter-Transistor 343 und einem dritten p-MOS-Inverter-Transistor 344. Der Ausgang 107 ist mit den Gate-Anschlüssen der Transistoren 344, 343 gekoppelt. Ein erster Source-/Drain-Anschluss des dritten p-MOS-Inverter-Transistors 344 ist auf das elektrische Versorgungspotential 307 gebracht. Der zweite Source-/Drain-Anschluss des dritten p-MOS-Inverter-Transistors 344 ist mit einem ersten Source-/Drain-Anschluss des dritten n-MOS-Inverter-Transistors 343 gekoppelt, dessen zweiter Source-/Drain-Anschluss auf das elektrische Massepotential 308 gebracht ist. Der zweite Source-/Drain-Anschluss des dritten p-MOS-Inverter-Transistors 344 und der erste Source-/Drain-Anschluss des dritten n-MOS-Inverter-Transistors 343 sind mit dem globalen Ausgang 345 gekoppelt.

[0090] Im Weiteren wird die Funktionalität der Logik-Schaltkreis-Anordnung 300 beschrieben.

[0091] Die Signalpfad-Einheit 309 führt anschaulich die Logikoperationen mit den Eingabesignalen  $\overline{a_0}$ ,  $\overline{a_1}$ ,  $a_0$  und  $a_1$  durch. Die Funktionalität von Gleichung (20) wird anschaulich von den in der Fig. 3 gezeigten Weise verschalteten p-MOS-Transistoren 314 bis 325 durchgeführt, wohingegen die Logikoperation gemäß Gleichung (21) anschaulich durch die n-MOS-Transistoren 326 bis 337 durchgeführt wird. Zwischen den Datensig-

nal-Eingängen 101, 102 und den Signalpfad-Eingängen 310 bis 313 sind die beiden Inverter-Schaltkreise 301, 304 vorgesehen, um die komplementären Signale  $\overline{a_0}$ ,  $\overline{a_1}$  zu generieren. Zwischen Ausgang 107 und dem globalen Ausgang 345 ist der treibende Inverter-Schaltkreis 342 geschaltet. Für die Zelle mit zwei Eingängen 101, 102 werden gemäß dem beschriebenen Ausführungsbeispiel dreißig Transistoren benötigt, nämlich die zwölf p-MOS-Logik-Transistoren 314 bis 325, die zwölf n-MOS-Logik-Transistoren 326 bis 337 und die sechs Transistoren 302, 303, 306, 343, 344. Die Anzahl der Konfigurationsbits bzw. der Logikfunktionssignale ist vier ( $c_0, c_1, c_2, c_3$ ).

[0092] Es ist anzumerken, dass anstelle von vier gemeinsamen Logikfunktionssignalen  $c_0$  bis  $c_3$  alternativ auch die zwölf p-MOS-Logik-Transistoren 314 bis 325 mit vier separaten Logikfunktionssignalen betrieben werden können und die zwölf n-MOS-Logik-Transistoren 326 bis 337 mit vier separaten, von  $c_0$  bis  $c_3$  unterschiedlichen Logikfunktionssignalen betrieben werden können.

[0093] Die anhand von **Fig. 3** dargestellte Realisierung, welche auf zwei Datensignale  $a_0, a_1$  bezogen ist, lässt sich beliebig für drei, vier, fünf, ..., n Eingänge durchführen. Dann vergrößert sich jedoch die Anzahl der seriell angeordneten Transistoren in einem Produktterm. Ferner kann die Verarbeitungsgeschwindigkeit der Zelle aufgrund einer erhöhten kapazitiven Last an  $y_{int}$  reduziert werden. In ungünstigen Szenarien kann aufgrund einer abgeschwächten Flanke von  $y_{int}$  die Störsicherheit der Zelle leicht herabgesetzt sein.

[0094] Im Weiteren wird bezugnehmend auf **Fig. 4** eine Logik-Schaltkreis-Anordnung 400 gemäß einem dritten Ausführungsbeispiel der Erfindung beschrieben, bei der eine Logik-Verknüpfung mit  $n=3$  Eingangssignalen bei einer hohen Verarbeitungsgeschwindigkeit und einer besonders guten Störsicherheit erreicht ist.

[0095] Die Logik-Schaltkreis-Anordnung 400 weist zusätzlich zu den ersten und zweiten Datensignal-Eingängen 101, 102, an denen die ersten und zweiten Datensignale  $a_0, a_1$  bereitgestellt sind, einen dritten Datensignal-Eingang 401 auf, an dem ein drittes Datensignal  $a_2$  bereitgestellt ist. Bevor die Datensignale  $a_0, a_1$  und  $a_2$  sowie ihre Logik-Inversen  $\overline{a_0}$ ,  $\overline{a_1}$  und  $\overline{a_2}$  in erste und zweite Signalpfad-Einheiten 405, 406 eingekoppelt werden, durchläuft jedes der Datensignale  $a_0, a_1$  und  $a_2$  einen jeweils zugehörigen Inverter-Schaltkreis 301, 304 bzw. 402 zum Generieren des jeweiligen logischen Inversen. Da in **Fig. 4** gegenüber **Fig. 3** ein zusätzlicher Datensignal-Eingang 401 vorgesehen ist, ist ein zugehöriger vierter Inverter-Schaltkreis 402 vorgesehen. Der dritte Datensignal-Eingang 401 ist mit den Gate-Anschlüssen des vierten n-MOS-Inverter-Transistors 403 und des vierten p-MOS-Inverter-Transistors 404 gekoppelt. Ein erster Source-/Drain-Anschluss des vierten p-MOS-Inverter-Transistors 404 ist auf das Versorgungspotential 307 gebracht. Der zweite Source-/Drain-Anschluss des vierten p-MOS-Inverter-Transistors 404 ist mit einem ersten Source-/Drain-Anschluss des vierten n-MOS-Inverter-Transistors 403 gekoppelt, dessen zweiter Source-/Drain-Anschluss auf das elektrische Massepotential 308 gebracht ist. Der zweite Source-/Drain-Anschluss des vierten p-MOS-Inverter-Transistors 404 und der erste Source-/Drain-Anschluss des vierten n-MOS-Inverter-Transistors 403 sind miteinander und mit einem Eingang eines Multiplexer-Schaltkreises 409 gekoppelt.

[0096] Bei der Logik-Schaltkreis-Anordnung 400 aus **Fig. 4** werden in eine erste Signalpfad-Einheit 405 die Signale  $\overline{a_0}$ ,  $\overline{a_1}$ ,  $a_0$  und  $a_1$  auf dieselbe Weise eingekoppelt wie oben bezugnehmend auf **Fig. 3** beschrieben. Der Aufbau und die Verschaltung innerhalb der ersten Signalpfad-Einheit 405 ist mit dem Aufbau und der Verschaltung in der Signalpfad-Einheit 309 aus **Fig. 3** identisch. Mittels der Logikfunktionssignale  $c_0$  bis  $c_3$  wird eine spezielle Logikfunktion vorgegeben bzw. ausgewählt, welche von den Transistoren 314 bis 337 der ersten Signalpfad-Einheit 405 erfüllt wird. Gemäß dieser ausgewählten Logikfunktion werden die Signale  $\overline{a_0}$ ,  $\overline{a_1}$ ,  $a_0$  und  $a_1$  verarbeitet und an einem ersten Signalpfad-Ausgang 407 bereitgestellt.

[0097] Eine zweite Signalpfad-Einheit 406 ist hinsichtlich Aufbau und Struktur identisch mit der ersten Signalpfad-Einheit 405.

[0098] An vier Signalpfad-Eingängen der zweiten Signalpfad-Einheit 406 sind wiederum die Datensignale  $\overline{a_0}$ ,  $\overline{a_1}$ ,  $a_0$  und  $a_1$  bereitgestellt. Diese Signale werden mittels einer Logikfunktion derart miteinander verknüpft, dass an einem zweiten Signalpfad-Ausgang 408 der zweiten Signalpfad-Einheit 406 ein entsprechendes Signal bereitgestellt wird. Der wesentliche Unterschied zwischen den Signalpfad-Einheiten 405, 406 besteht darin, dass die zweite Signalpfad-Einheit 406 eine Logikfunktion durchführt, welche mittels Bereitstellens fünfter bis achter Logikfunktionssignale  $c_4$  bis  $c_7$  vorgegeben wird. Mit anderen Worten können die Logikfunktionssignale  $c_4$  bis  $c_7$  von den Logikfunktionssignalen  $c_0$  bis  $c_3$  der ersten Signalpfad-Einheit 405 unterschiedlich sein, so dass die von den beiden Signalpfad-Einheiten 405, 406 realisierten Logikfunktionen voneinander unterschiedlich sein können.

[0099] Wie ferner in **Fig. 4** gezeigt, sind die Signalpfad-Ausgänge **407**, **408** der Signalpfad-Einheiten **405**, **406** mit Eingängen des Multiplexer-Schaltkreises **409** gekoppelt. Somit sind dem Multiplexer-Schaltkreis **409** als Eingabesignale die Datensignale  $\overline{a_2}$ ,  $a_2$  sowie die beiden Ausgangssignale der Signalpfad-Einheiten **405**, **406** (d.h. die Ergebnisse der gemäß den Logikfunktionen der Signalpfad-Einheiten **405**, **406** verarbeiteten Signale  $\overline{a_0}$ ,  $\overline{a_1}$ ,  $a_0$  und  $a_1$ ) als Eingabesignale bereitgestellt. Diese vier Eingabesignale werden von dem Multiplexer-Schaltkreis **409** weiterverarbeitet.

[0100] Der Multiplexerschaltkreis **409** ist gebildet aus einem ersten p-MOS-Multiplexer-Transistor **410**, einem ersten n-MOS-Multiplexer-Transistor **411**, einem zweiten p-MOS-Multiplexer-Transistor **412** und einem zweiten n-MOS-Multiplexer-Transistor **413**. Der erste Signalpfad-Ausgang **407** ist mit ersten Source-/Drain-Anschlüssen des ersten p-MOS-Multiplexer-Transistors **410** und des ersten n-MOS-Multiplexer-Transistors **411** gekoppelt. Der Gate-Anschluss des ersten p-MOS-Multiplexer-Transistors **410** ist mit einem Ausgang des vierten p-MOS-Inverter-Transistors **404** gekoppelt. Der zweite Signalpfad-Ausgang **408** ist mit ersten Source-/Drain-Anschlüssen des zweiten p-MOS-Multiplexer-Transistors **412** und des zweiten n-MOS-Multiplexer-Transistors **413** gekoppelt. An den Gate-Anschluss des zweiten p-MOS-Multiplexer-Transistors **412** ist das dritte Datensignal  $a_2$  angelegt. Der Gate-Anschluss des zweiten p-MOS-Multiplexer-Transistors **412** mit dem Gate-Anschluss des ersten n-MOS-Multiplexer-Transistors **411** gekoppelt. Ferner ist der Gate-Anschluss des zweiten p-MOS-Multiplexer-Transistors **413** mit dem Gate-Anschluss des ersten p-MOS-Multiplexer-Transistors **410** gekoppelt. Die zweiten Source-/Drain-Anschlüsse der Transistoren **410** bis **413** sind miteinander und mit einem Eingang des dritten Inverter-Schaltkreises **342** gekoppelt.

[0101] Im Weiteren wird die Funktionalität der Logik-Schaltkreis-Anordnung **400** näher beschrieben.

[0102] Die in **Fig. 4** gezeigte Verschaltung basiert auf den folgenden mathematischen Überlegungen. Gemäß der Bool'schen Algebra gilt:

$$y(a_2, a_1, a_0) = \overline{a_2} \cdot y_0(a_1, a_0) \vee a_2 \cdot y_1(a_1, a_0) \quad (23)$$

[0103]  $y_0$  ist gemäß Gleichung (20) definiert und hängt von den Variablen  $c_0, c_1, c_2, c_3$  ab.  $y_1$  hingegen wird mit den zusätzlichen Schaltvariablen oder Logikfunktionsvariablen  $c_4, c_5, c_6, c_7$  konfiguriert und wird gemäß folgender Vorschrift gebildet:

$$y_1 = \overline{c_4} \cdot \overline{a_1} \cdot \overline{a_0} \vee \overline{c_5} \cdot \overline{a_1} \cdot a_0 \vee \overline{c_6} \cdot a_1 \cdot \overline{a_0} \vee \overline{c_7} \cdot a_1 \cdot a_0 \quad (24)$$

[0104] Die Ausgangssignale der beiden Teilfunktionen  $y_0$  und  $y_1$  werden mittels des Multiplexer-Schaltkreises **409** zusammengeführt, welcher mittels des Datensignals  $a_2$  gesteuert wird (vergleiche **Fig. 4**).

[0105] Es ist anzumerken, dass der Multiplexer-Schaltkreis **409** je nach Anwendungsbedürfnissen beliebig implementiert werden kann, zum Beispiel unter Verwendung von Logik-Gattern oder n-Kanal-Passgates.

[0106] Die allgemeine Bildungsvorschrift für eine Universalzelle-Logikfunktion mit n Eingängen lautet:

$$y(a_{n-1}, \dots, a_1, a_0) = \overline{a_{n-1}} \dots \overline{a_2} \cdot y_0 \vee \overline{a_{n-1}} \dots a_2 \cdot y_1 \vee \dots \vee a_{n-1} \dots a_2 \cdot y_{2^{(n-2)}-1} \quad (25)$$

[0107] Die Realisierung von Zellen mit  $n > 3$  erfolgt in ähnlicher Weise wie in **Fig. 4**, wobei der Multiplexer-Schaltkreis **409** zwei- oder mehrstufig ausgeführt werden kann, zum Beispiel mit einer Baumstruktur. Die Anzahl der erforderlichen Konfigurationsvariablen ist  $2^n$ .

[0108] Eine andere Möglichkeit der Skalierung beruht auf dem Clustern mehrerer einfacher Zellen. Dieses Clustern bzw. gemeinsame Anordnen kann unter Verwendung einer Baumstruktur oder unter Verwendung einer Kaskadenstruktur erfolgen.

[0109] Im Weiteren wird beschrieben, wie die erfindungsgemäße universelle Logikzelle in Form der Logik-Schaltkreis-Anordnung in einem FPGA eingesetzt werden kann, wodurch eine interessante Alternative zu der LUT-Realisierung gemäß dem Stand der Technik ermöglicht ist.

[0110] Basis einer FPGA-Architektur ist eine kombinatorische Logikzelle. In der Praxis erweisen sich insbesondere diejenigen Bausteine als interessant, deren Kombinatorik vollständig alle Logik-Funktionen der Ein-

gänge bereitstellt. Nicht direkt realisierbare Logik-Funktionen stellen im FPGA-Design-Flow ein erhebliches Mapping-Problem dar. Unter den FPGAs mit vollständiger Logik-Funktion sind Realisierungen mittels Look-Up-Tabellen möglich. Rekonfigurierbarkeit gewinnen diese Architekturen, indem die Funktionswerte aus Speicher- oder Latchzellen gelesen werden, die jederzeit neu belegt werden können.

**[0111]** Die universelle Logikzelle der Logik-Schaltkreis-Anordnung **300** bzw. **400** gemäß der Erfindung ist als FPGA-Kombinatorik verwendbar, wenn die Konfigurationsvariablen  $c_i$  zum Beispiel in eine Speicherzelle geschrieben werden. Das Prinzip der erfindungsgemäßen Realisierung der kombinatorischen Funktionen ist allerdings grundsätzlich von dem der Look-Up-Tabelle unterschiedlich. Erfindungsgemäß wird die Logikfunktion durch die Konfigurationsvariablen  $c_i$  eindeutig festgelegt.

**[0112]** Verglichen mit bekannten Architekturen (zum Beispiel XILINX, ACTEL) weist die universelle Logikzelle der Erfindung mindestens gleich gute Werte hinsichtlich Zeiteigenschaften und Leistungsaufnahme auf. Allerdings ist ihre hervorragende Skalierbarkeit und größere Störsicherheit gegenüber den aus dem Stand der Technik bekannten Lösungen stark verbessert. Gatterbasierte Ansätze, die ausreichend skalierbar und stör sicher sind, sind weniger kompakt bzw. weisen eine deutlich größere Verzögerungszeit auf.

**[0113]** Ein besonders interessantes Einsatzgebiet der erfindungsgemäßen Logik-Schaltkreis-Anordnung ist ihre Verwendung als universelle Logikzelle als Kombinatorik in einer "Structured ASICs"-Struktur.

**[0114]** In Zusammenhang mit den hohen Kosten für halbleitertechnologische Masken, die in zukünftigen Technologie-Generationen weiter ansteigen werden, gewinnen maskenprogrammierte "Structured ASICs" zunehmend an Bedeutung. Hierunter wird der Ansatz verstanden, einen ASIC, ähnlich wie ein FPGA, mit einem regelmäßigen Feld von Zellen und Interconnect-Ressourcen zu versehen. Flexible Rekonfigurierbarkeit ist für einen "Structured ASIC" entbehrlich, so dass die (Logik-)Funktion der Zellen und das Interconnect nach dem Entwurf des Chips fest eingestellt werden können. Bei der Konfigurierung eines "Structured-ASIC" sind wenige Maskenebenen ausreichend, im Idealfall ist eine einzige Maske ausreichend. Abgesehen von der Einsparung fast der gesamten Maskenkosten bewirkt die sehr regelmäßige Struktur, dass Effekte im Tief-Submikrometerbereich ("Deep Submicron") besser modellierbar und beherrschbar werden. Diesen Vorteilen einer "Structured ASICs"-Struktur stehen gemäß dem Stand der Technik jedoch eine suboptimale Flächennutzung, eine höhere Leistungsaufnahme und eine geringe maximale Taktfrequenz gegenüber, verglichen mit custom ASIC auf der Basis von Standardzellen.

**[0115]** Als Kombinatorik in einem "Structured ASIC" verstärkt die erfindungsgemäße Logik-Schaltkreis-Anordnung die Vorzüge einer "Structured ASIC"-Struktur in besonderem Maße. Bei geeigneter Implementierung können zusätzliche vorteilhafte Effekte erreicht werden. Eine feste Konfiguration einer solchen universellen Logikzelle als Ausgestaltung der erfindungsgemäßen Logik-Schaltkreis-Anordnung kann erreicht werden, indem die Schaltvariablen  $c_i$  fest auf Logik-Werte "0" oder "1" gelegt werden. In diesem Falle ist entweder der Produktterm im Pfad der p-MOS-Transistoren dauerhaft aktiviert und der gegengleiche Produktterm im Pfad der n-MOS-Transistoren dauerhaft abgeschaltet oder umgekehrt. In jedem Fall können die Transistoren an den Schaltvariablen eingespart und durch eine feste Verbindung ersetzt werden, zum Beispiel unter Verwendung von Vias oder anderen Kontaktierungs- bzw. Kurzschluss-Elementen.

**[0116]** Im Weiteren wird bezugnehmend auf **Fig. 5** eine Layout-Draufsicht einer Mehrschicht-Anordnung von Halbleiterkomponenten beschrieben, welche für eine "Structured ASIC"-Struktur der erfindungsgemäßen Logik-Schaltkreis-Anordnung verwendet werden kann.

**[0117]** In Draufsicht **500** sind einzelne Produktterme gemäß der boolschen Funktion gezeigt. Ferner sind Vias, das heißt senkrecht zu der Papierebene zu **Fig. 5** verlaufende elektrisch leitfähige Verbindungen mit Bezugszeichen **501** gezeigt. Darüber hinaus sind eine erste Metallebene **502** und eine zweite Metallebene **503** gezeigt, in welchen Leitungen geführt sind, mittels welchen Kopplungen zwischen den Einzelstrukturen der erfindungsgemäßen Logik-Schaltkreis-Anordnung realisiert sind.

**[0118]** Das Einsparen von Transistoren an den Schaltvariablen und deren Ersetzung durch eine feste Verbindung kann erreicht werden, indem  $y_{int}$  gemäß **Fig. 5** in der zweiten Metallebene **503** geführt wird. Es überlappen kleine Buchten der zweiten Metallebene **503** mit Zuleitungen in der ersten Metallebene **502**. Die Überlappungen haben vorzugsweise im Wesentlichen die Größe eines Vias **501**.

**[0119]** Anschaulich ist aus **Fig. 5** die Ausgestaltung der konfigurierbaren universalen Logikzelle, das heißt der erfindungsgemäßen Logik-Schaltkreis-Anordnung, in eine "Structured ASIC"-Struktur ersichtlich.



[0120] Bezugnehmend auf Fig. 5 ist ferner anzumerken, dass jeweils nicht mit einem Via nach  $y_{int}$  hin verbundene Produktterme auch nicht mit der Versorgungsspannung bzw. mit dem Massepotential verbunden werden. Dagegen sind aktivierte Produktterme jeweils unter Verwendung eines Vias an die Versorgungsspannung bzw. an das Massepotential angeschlossen.

[0121] Die Anzahl der benötigten Transistoren zur Realisierung der universellen Logikzelle für zwei Eingänge  $a_0, a_1$  sinkt in diesem Fall von ursprünglich dreißig (Fig. 3) auf zweiundzwanzig Transistoren. Dies ist aus Fig. 6 ersichtlich, wo eine Logik-Schaltkreis-Anordnung 600 als Structured-ASIC dargestellt ist. An einer konfigurierbaren Logik-Funktion (in Fig. 6 ist ein XOR2-Gatter gezeigt) sind aktiv sogar nur vierzehn Transistoren beteiligt, wohingegen die restlichen acht Transistoren inaktiv sind. Simultan sinkt die Last an  $y_{int}$  stark ab, z.B. auf ungefähr die Hälfte. Somit ist eine kurze Verzögerungszeit der Zelle und eine sehr gute Realisierungseffizienz mit der fest konfigurierten universellen Logikzelle der Erfindung erreichbar. Bei der in Fig. 6 gezeigten Konfiguration der erfindungsgemäßen Logik-Schaltkreis-Anordnung ist das Bereitstellen variabler Logikfunktionssignale  $c_i$  entbehrlich und durch eine festverdrahtete Lösung ersetzt.

[0122] Im Weiteren wird der Aufbau der Logik-Schaltkreis-Anordnung 600 gemäß einem vierten Ausführungsbeispiel der Erfindung beschrieben.

[0123] Die Logik-Schaltkreis-Anordnung 600 ist als festkonfiguriertes XOR2-Gatter ausgeführt.

[0124] Es sind im Unterschied zu der Logik-Schaltkreis-Anordnung 300 bei der Logik-Schaltkreis-Anordnung 600 keine Logikfunktions-Eingänge 338 bis 341 vorgesehen, und es müssen keine Logikfunktionssignale bereitgestellt werden. Das Generieren der inversen Datensignale  $\overline{a_0}, \overline{a_1}$  erfolgt gemäß Fig. 6 wie gemäß Fig. 3 unter Verwendung der ersten und zweiten Inverter-Teilschaltkreise 301, 304. Die Weiterverarbeitung des Ausgangssignals  $y_{int}$  an Ausgang 107 erfolgt in Fig. 6 wie in Fig. 3 unter Verwendung des dritten Inverter-Schaltkreises 342, so dass an einem globalen Ausgang 345 ein Ausgangssignal  $y_0$  bereitgestellt wird.

[0125] Ein wesentlicher Unterschied zwischen der Logik-Schaltkreis-Anordnung 300 und der Logik-Schaltkreis-Anordnung 600 besteht darin, dass bei der Signalpfad-Einheit 601 gegenüber der Signalpfad-Einheit 309 einige Transistoren eingespart sind.

[0126] Der erste Signalpfad-Eingang 310 ist mit den Gate-Anschlüssen des sechsten p-MOS-Logik-Transistors 319, des siebten n-MOS-Logik-Transistors 332 und des vierten p-MOS-Logik-Transistors 317 gekoppelt. Der erste Signalpfad-Eingang 310 ist ferner mit dem Gate-Anschluss des fünften n-MOS-Logik-Transistors 330 gekoppelt. Der zweite Signalpfad-Eingang 311 ist mit den Gate-Anschlüssen des neunten n-MOS-Logik-Transistors 334, des zehnten n-MOS-Logik-Transistors 335, des dritten p-MOS-Logik-Transistors 316 und des achten p-MOS-Logik-Transistors 321 gekoppelt. Ferner ist der dritte Signalpfad-Eingang 312 mit den Gate-Anschlüssen des fünften p-MOS-Logik-Transistors 318, des sechsten n-MOS-Logik-Transistors 331 und des siebten p-MOS-Logik-Transistors 320 gekoppelt. Ferner ist der dritte Signalpfad-Eingang 312 mit dem Gate-Anschluss des achten n-MOS-Logik-Transistors 333 gekoppelt. Der vierte Signalpfad-Eingang 313 ist mit den Gate-Anschlüssen des ersten p-MOS-Logik-Transistors 314, des zweiten p-MOS-Logik-Transistors 315, des elften n-MOS-Logik-Transistors 336 und des zwölften n-MOS-Logik-Transistors 337 gekoppelt.

[0127] Erste Source-/Drain-Anschlüsse der ersten bis vierten p-MOS-Logik-Transistoren 314 bis 317 sind auf dem Potential der Versorgungsspannung 307. Der zweite Source-/Drain-Anschluss des ersten p-MOS-Logik-Transistors 314 ist mit dem einen ersten Source-/Drain-Anschluss des fünften p-MOS-Logik-Transistors 318 gekoppelt, dessen zweiter Source-/Drain-Anschluss mit einem ersten Source-/Drain-Anschluss des neunten n-MOS-Logik-Transistors 332 gekoppelt ist, dessen zweiter Source-/Drain-Anschluss mit dem Ausgang 107 gekoppelt ist. Der zweite Source-/Drain-Anschluss des zweiten p-MOS-Logik-Transistors 315 ist mit einem ersten Source-/Drain-Anschluss des sechsten p-MOS-Logik-Transistors 319 gekoppelt, dessen zweiter Source-/Drain-Anschluss keine Kopplung aufweist. Der dritte p-MOS-Logik-Transistor 316 ist mit seinem zweiten Source-/Drain-Bereich mit einem ersten Source-/Drain-Bereich des siebten p-MOS-Logik-Transistors 320 gekoppelt, dessen zweiter Source-/Drain-Bereich keine Kopplung aufweist. Der zweite Source-/Drain-Bereich des vierten p-MOS-Logik-Transistors 317 ist mit einem ersten Source-/Drain-Bereich des achten p-MOS-Logik-Transistors 321 gekoppelt, dessen zweiter Source-/Drain-Bereich mit dem Ausgang 107, dem ersten Source-/Drain-Bereich des sechsten n-MOS-Logik-Transistors 331 und einem ersten Source-/Drain-Bereich des siebten n-MOS-Logik-Transistors 332 gekoppelt ist.

[0128] Ein erster Source-/Drain-Bereich des fünften n-MOS-Logik-Transistors 330 weist keine Kopplung auf. Der zweite Source-/Drain-Bereich des fünften n-MOS-Logik-Transistors 330 ist mit einem ersten Sour-

ce-/Drain-Bereich des neunten n-MOS-Logik-Transistors 334 gekoppelt, dessen zweiter Source-/Drain-Bereich auf das elektrische Massepotential 308 gebracht ist. Der zweite Source-/Drain-Bereich des sechsten n-MOS-Logik-Transistors 331 ist mit einem ersten Source-/Drain-Bereich des zehnten n-MOS-Logik-Transistors 335 gekoppelt, dessen zweiter Source-/Drain-Bereich auf das elektrische Massepotential 308 gebracht ist. Der zweite Source-/Drain-Bereich des siebten n-MOS-Logik-Transistors 332 ist mit einem ersten Source-/Drain-Bereich des elften n-MOS-Logik-Transistors 336 gekoppelt, dessen zweiter Source-/Drain-Bereich auf das elektrische Massepotential 308 gebracht ist.

[0129] Ein erster Source-/Drain-Bereich des achten n-MOS-Logik-Transistors 333 weist keine Kopplung auf. Der zweite Source-/Drain-Bereich des achten n-MOS-Logik-Transistors 333 ist mit einem ersten Source-/Drain-Bereich des zwölften n-MOS-Logik-Transistors 337 gekoppelt, dessen zweiter Source-/Drain-Bereich auf das elektrische Massepotential 308 gebracht ist.

[0130] Es ist anzumerken, dass bei der Logik-Schaltkreis-Anordnung 600 lediglich die Peripherietransistoren 302, 303, 305, 306, 343, 344 sowie die Logik-Transistoren 314, 317, 318, 321, 331, 332, 335 und 336 funktionell aktiv sind, wohingegen die restlichen in Fig. 6 gezeigten Transistoren inaktiv sind.

[0131] Im Weiteren wird bezugnehmend auf Fig. 7 eine Logik-Schaltkreis-Anordnung 700 gemäß einem fünften Ausführungsbeispiel der Erfindung beschrieben.

[0132] Bei der Logik-Schaltkreis-Anordnung 700 sind an einem ersten Datensignal-Eingang 101 und an einem zweiten Datensignal-Eingang 102 erste und zweite Datensignale  $a_0$ ,  $a_1$  bereitgestellt. Nach Durchlaufen einer Komplementärsignalgenerier-Einheit 702, die im Wesentlichen aus den gemäß Fig. 3 verschalteten ersten und zweiten Inverter-Schaltkreisen 301, 304 gebildet sein kann, sind abgesehen von den ersten und zweiten Datensignalen  $a_0$ ,  $a_1$  auch deren logisch komplementäre Signale an Datensignaleingängen einer Signalfad-Einheit 103 bereitgestellt. Im Unterschied zu der Logik-Schaltkreis-Anordnung 100 ist bei der Logik-Schaltkreis-Anordnung 700 genau eine Signalfad-Einheit 103 aus n-MOS-Feldeffekttransistoren 104 vorgesehen und die in Fig. 1 gezeigte Signalfad-Einheit 105 aus p-MOS-Feldeffekttransistoren 106 eingespart. An vier Steuereingängen 702 sind Logikfunktionssignale  $c_0$ ,  $c_1$ ,  $c_2$ ,  $c_3$  bereitgestellt. Mittels der Logikfunktionssignale an den Steuereingängen 702 ist ein vorgebbarer, von den Transistoren 104 gebildeter Signallaufpfad geschaltet, welcher Signallaufpfad mit der realisierten Logikfunktion korrespondiert. Ein Ausgabesignal, das die Verarbeitung der Datensignale  $a_0$ ,  $a_1$  gemäß der ausgewählten Logikfunktion darstellt, ist an einem Ausgang der Signalfad-Einheit 105 bereitgestellt, welcher Ausgang mit einem ersten Source-/Drain-Bereich eines Evaluierungs-Transistors 704 gekoppelt ist. Bei einem entsprechenden Signal an einem mit dem Gate-Bereich des Evaluierungs-Transistors 704 gekoppelten Evaluier-Eingang 706 liegt an einem mit dem zweiten Source-/Drain-Bereich des Evaluierungs-Transistors 704 gekoppelten Ausgang 107 der Logik-Schaltkreis-Anordnung 700 das verarbeitete Ausgabesignal an. Der zweite Source-/Drain-Bereich des Evaluierungs-Transistors 704 ist mit einem ersten Source-/Drain-Bereich eines Vorlade-Transistors 703 gekoppelt, dessen zweiter Source-/Drain-Bereich auf das elektrische Versorgungspotential 307 gebracht ist. Bei einem entsprechenden Signal an einem mit dem Gate-Bereich des Vorlade-Transistors 703 gekoppelten Vorlade-Eingang 705 liegt an dem mit dem ersten Source-/Drain-Bereich des Vorlade-Transistors 703 gekoppelten Ausgang 107 der Logik-Schaltkreis-Anordnung 700 das elektrische Versorgungspotential 307 als Referenz-Potential an.

[0133] Verglichen mit Fig. 1 ist in Fig. 7 somit der Pfad aus p-MOS-Transistoren eingespart. Das Pull-Down-Netzwerk 103 ist wie in Fig. 1 aus n-MOS-Transistoren gebildet, wohingegen in Fig. 7 das Pull-Up-Netzwerk 105 aus p-MOS-Transistoren eingespart ist und durch einen statisch oder dynamisch gesteuerten Precharge-Transistor 703 ersetzt ist. Alternativ kann auch der Signalfad aus n-MOS-Transistoren in Fig. 1 eingespart und durch einen Vorlade-Transistor ersetzt werden, wobei in diesem Fall der Signalfad aus p-MOS-Transistoren vorgesehen ist.

[0134] In diesem Dokument sind folgende Veröffentlichungen zitiert:

- [1] Wannemacher, M "Das FPGA-Kochbuch", Abb. 6.4: SRAM-Zelle von XILINX, 1. Auflage, International Thomson Publishing Company, Bonn, 1998, S 111
- [2] Wannemacher, M "Das FPGA-Kochbuch", Abb. 7.36: Logikblock (CLB) der XC4000-Familien, 1. Auflage, International Thomson Publishing Company, Bonn, 1998, 5.197
- [3] US 6,529,040 B1

Bezugszeichenliste

100	Logik-Schaltkreis-Anordnung
101	erster Datensignal-Eingang
102	zweiter Datensignal-Eingang
103	erste Signalpfad-Einheit
104	n-MOS-Transistoren
105	zweite Signalpfad-Einheit
106	p-MOS-Transistoren
107	Ausgang
108	Weiterverarbeitungs-Einheit
109	globaler Ausgang
200	Tabelle
300	Logik-Schaltkreis-Anordnung
301	erster Inverter-Schaltkreis
302	erster n-MOS-Inverter-Transistor
303	erster p-MOS-Inverter-Transistor
304	zweiter Inverter-Schaltkreis
305	zweiter n-MOS-Inverter-Transistor
306	zweiter p-MOS-Inverter-Transistor
307	Versorgungspotential
308	Massepotential
309	Signalpfad-Einheit
310	erster Signalpfad-Eingang
311	zweiter Signalpfad-Eingang
312	dritter Signalpfad-Eingang
313	vierter Signalpfad-Eingang
314	erster p-MOS-Logik-Transistor
315	zweiter p-MOS-Logik-Transistor
316	dritter p-MOS-Logik-Transistor
317	vierter p-MOS-Logik-Transistor
318	fünfter p-MOS-Logik-Transistor
319	sechster p-MOS-Logik-Transistor
320	siebter p-MOS-Logik-Transistor
321	achter p-MOS-Logik-Transistor
322	neunter p-MOS-Logik-Transistor
323	zehnter p-MOS-Logik-Transistor
324	elfter p-MOS-Logik-Transistor
325	zwölfter p-MOS-Logik-Transistor
326	erster n-MOS-Logik-Transistor
327	zweiter n-MOS-Logik-Transistor
328	dritter n-MOS-Logik-Transistor
329	vierter n-MOS-Logik-Transistor
330	fünfter n-MOS-Logik-Transistor
331	sechster n-MOS-Logik-Transistor
332	siebter n-MOS-Logik-Transistor
333	achter n-MOS-Logik-Transistor
334	neunter n-MOS-Logik-Transistor
335	zehnter n-MOS-Logik-Transistor
336	elfter n-MOS-Logik-Transistor
337	zwölfter n-MOS-Logik-Transistor
338	erster Logikfunktions-Eingang
339	zweiter Logikfunktions-Eingang
340	dritter Logikfunktions-Eingang
341	vierter Logikfunktions-Eingang
342	dritter Inverter-Schaltkreis
343	dritter n-MOS-Inverter-Transistor
344	dritter p-MOS-Inverter-Transistor
345	globaler Ausgang
400	Logik-Schaltkreis-Anordnung

401	dritter Datensignal-Eingang
402	vierter Inverter-Schaltkreis
403	vierter n-MOS-Inverter-Transistor
404	vierter p-MOS-Inverter-Transistor
405	erste Signalpfad-Einheit
406	zweite Signalpfad-Einheit
407	erster Signalpfad-Ausgang
408	zweiter Signalpfad-Ausgang
409	Multiplexer-Schaltkreis
410	erster p-MOS-Multiplexer-Transistor
411	erster n-MOS-Multiplexer-Transistor
412	zweiter p-MOS-Multiplexer-Transistor
413	zweiter n-MOS-Multiplexer-Transistor
500	Draufsicht
501	Vias
502	erste Metallebene
503	zweite Metallebene
600	Logik-Schaltkreis-Anordnung
601	Signalpfad-Einheit
700	Logik-Schaltkreis-Anordnung
701	Komplementärsignalgenerier-Einheit
702	Steuereingänge
703	Vorlade-Transistor
704	Evaluier-Transistor
705	Vorlade-Eingang
706	Evaluier-Eingang

#### Patentansprüche

##### 1. Logik-Schaltkreis-Anordnung,

- mit mindestens zwei Datensignal-Eingängen, an denen mindestens zwei Datensignale bereitstellbar sind;
- mit einem mit den Datensignal-Eingängen gekoppelten ersten Signalpfad mit einer Mehrzahl von Transistoren eines ersten Leitungstyps, wobei die Transistoren derart miteinander verschaltbar sind, dass sie eine erste Logikfunktion von allen möglichen Logikfunktionen zur logischen Verknüpfung der zwei Datensignale realisieren, so dass ein das Ergebnis der ersten Logikfunktion repräsentierendes Ausgangssignal bereitgestellt wird;
- mit einer Mehrzahl von mit den Transistoren gekoppelten Steuereingängen, mittels welcher ein vorgebbarer, von den Transistoren gebildeter Signallaufpfad zu- oder abschaltbar ist.

2. Logik-Schaltkreis-Anordnung nach Anspruch 1, mit einem Evaluierungs-Schalter und mit einem Vorlade-Schalter, welche Schalter derart verschaltet und steuerbar sind, dass an einem Ausgang der Logik-Schaltkreis-Anordnung bei geöffneten Evaluierungs-Schalter und geschlossenem Vorlade-Schalter das Ausgabesignal bereitgestellt ist, und dass an dem Ausgang der Logik-Schaltkreis-Anordnung bei geöffnetem Vorlade-Schalter und geschlossenem Evaluierungs-Schalter ein Referenzsignal bereitgestellt ist.

3. Logik-Schaltkreis-Anordnung nach Anspruch 2, bei dem der Evaluierungs-Schalter und der Vorlade-Schalter jeweils Transistoren sind.

4. Logik-Schaltkreis-Anordnung nach Anspruch 1, mit einem mit den Datensignal-Eingängen gekoppelten zweiten Signalpfad mit einer Mehrzahl von Transistoren eines zu dem ersten Leitungstyp komplementären zweiten Leitungstyps, wobei die Transistoren derart miteinander verschaltbar sind, dass sie eine zweite Logikfunktion von allen möglichen Logikfunktionen zur logischen Verknüpfung der zwei Datensignale realisieren, so dass ein das Ergebnis der zweiten Logikfunktion repräsentierendes Ausgangssignal bereitgestellt wird, wobei das Ergebnis der zweiten Logikfunktion invers zu dem Ergebnis der ersten Logikfunktion ist.

##### 5. Logik-Schaltkreis-Anordnung nach Anspruch 4,

bei der zumindest ein Teil der ersten und/oder der zweiten Transistoren

- Feldeffekttransistoren; oder
  - Bipolartransistoren
- sind.

6. Logik-Schaltkreis-Anordnung nach Anspruch 4 oder 5, bei welcher

- der erste Leitungstyp der p-Leitungstyp und der zweite Leitungstyp der n-Leitungstyp ist; oder
- der erste Leitungstyp der n-Leitungstyp und der zweite Leitungstyp der p-Leitungstyp ist.

7. Logik-Schaltkreis-Anordnung nach einem der Ansprüche 4 bis 6, eingerichtet als CMOS-Schaltkreis-Anordnung.

8. Logik-Schaltkreis-Anordnung nach einem der Ansprüche 1 bis 7, eingerichtet als Application-Specific Integrated Circuit.

9. Logik-Schaltkreis-Anordnung nach einem der Ansprüche 1 bis 8, eingerichtet als

- Programmable Logic Device;
- Field-Programmable Gate-Array;
- maskenprogrammierter Application-Specific Integrated Circuit.

10. Logik-Schaltkreis-Anordnung nach einem der Ansprüche 1 bis 9, bei der die Logikfunktionen zumindest eine der folgenden Logikfunktionen aufweist:

- UND-Verknüpfung;
- ODER-Verknüpfung;
- Nicht-UND-Verknüpfung;
- Nicht-ODER-Verknüpfung;
- Exklusiv-ODER-Verknüpfung;
- Exklusiv-Nicht-ODER-Verknüpfung.

11. Logik-Schaltkreis-Anordnung nach einem der Ansprüche 4 bis 10, bei der zumindest einem Teil der Transistoren an den Steuereingängen mindestens ein Logikfunktionssignal zum Vorgeben der ersten und zweiten Logikfunktion unveränderlich vorgegeben ist.

12. Logik-Schaltkreis-Anordnung nach Anspruch 11, mit einer mit zumindest einem Teil der Transistoren gekoppelten Speicher-Einrichtung, in welcher die Werte des mindestens einen Logikfunktionssignals speicherbar sind.

13. Logik-Schaltkreis-Anordnung nach einem der Ansprüche 4 bis 12, bei der zumindest einem Teil der Transistoren mindestens ein Logikfunktionssignal zum Vorgeben der ersten und zweiten Logikfunktion variabel vorgebbbar ist.

14. Logik-Schaltkreis-Anordnung nach einem der Ansprüche 4 bis 13, bei der zwischen den mindestens zwei Datensignal-Eingängen einerseits und den ersten und zweiten Signalpfaden andererseits eine Einrichtung zum Generieren des zu einem jeweiligen Datensignal logisch komplementären Datensignals angeordnet ist.

15. Logik-Schaltkreis-Anordnung nach Anspruch 14, bei der die Einrichtung für jedes der Datensignale einen Inverter-Teilschaltkreis aufweist.

16. Logik-Schaltkreis-Anordnung nach einem der Ansprüche 4 bis 15, bei der den ersten und zweiten Signalpfaden eine Ausgangssignalweiterverarbeitungs-Einrichtung nachgeschaltet ist.

17. Logik-Schaltkreis-Anordnung nach Anspruch 16 bei der die Ausgangssignalweiterverarbeitungs-Einrichtung

- im Falle von mindestens zwei Datensignal-Eingängen einen Inverter; und/oder
- im Falle von drei oder mehr Datensignal-Eingängen eine Multiplexer-Einrichtung aufweist.

18. Logik-Schaltkreis-Anordnung nach einem der Ansprüche 1 bis 17, eingerichtet zum Verarbeiten von digitalen Datensignalen und/oder digitalen Logikfunktionssignalen.

19. Logik-Schaltkreis-Anordnung nach einem der Ansprüche 4 bis 18, bei der die Transistoren verschaltet sind zum Verarbeiten von zwei Datensignalen  $a_0$ ,  $a_1$  zu einem Ausgangssignal  $y(a_0, a_1)$  unter Verwendung von

Logikfunktionssignalen  $c_0, c_1, c_2, c_3$  mit  $c_i \in \{1, 0\}$  zum Vorgeben der Logikfunktion gemäß folgender Abbildungsvorschrift:

$$y = \overline{c_0} \cdot \overline{a_1} \cdot \overline{a_0} \vee \overline{c_1} \cdot \overline{a_1} \cdot a_0 \vee \overline{c_2} \cdot a_1 \cdot \overline{a_0} \vee \overline{c_3} \cdot a_1 \cdot a_0$$

und

$$\overline{y} = c_0 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_1 \cdot \overline{a_1} \cdot a_0 \vee c_2 \cdot a_1 \cdot \overline{a_0} \vee c_3 \cdot a_1 \cdot a_0.$$

20. Logik-Schaltkreis-Anordnung nach einem der Ansprüche 4 bis 18, bei der die Transistoren verschaltet sind zum Verarbeiten von drei Datensignalen  $a_0, a_1, a_2$  zu einem Ausgangssignal  $y(a_0, a_1, a_2)$  unter Verwendung von Logikfunktionssignalen  $c_0, c_1, c_2, c_3, c_4, c_5, c_6, c_7$  mit  $c_i \in \{1, 0\}$  zum Vorgeben der Logikfunktion gemäß folgender Abbildungsvorschrift:

$$y(a_2, a_1, a_0) = \overline{a_2} \cdot y_0 \vee a_2 \cdot y_1$$

mit

$$y_0 = \overline{c_0} \cdot \overline{a_1} \cdot \overline{a_0} \vee \overline{c_1} \cdot \overline{a_1} \cdot a_0 \vee \overline{c_2} \cdot a_1 \cdot \overline{a_0} \vee \overline{c_3} \cdot a_1 \cdot a_0$$

und

$$\overline{y_0} = c_0 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_1 \cdot \overline{a_1} \cdot a_0 \vee c_2 \cdot a_1 \cdot \overline{a_0} \vee c_3 \cdot a_1 \cdot a_0$$

und

$$y_1 = \overline{c_4} \cdot \overline{a_1} \cdot \overline{a_0} \vee \overline{c_5} \cdot \overline{a_1} \cdot a_0 \vee \overline{c_6} \cdot a_1 \cdot \overline{a_0} \vee \overline{c_7} \cdot a_1 \cdot a_0$$

und

$$\overline{y_1} = c_4 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_5 \cdot \overline{a_1} \cdot a_0 \vee c_6 \cdot a_1 \cdot \overline{a_0} \vee c_7 \cdot a_1 \cdot a_0.$$

21. Logik-Schaltkreis-Anordnung nach einem der Ansprüche 4 bis 18, bei der die Transistoren verschaltet sind zum Verarbeiten von  $n$  Datensignalen  $a_0, a_1, \dots, a_{n-1}$  zu einem Ausgangssignal  $y(a_0, a_1, \dots, a_{n-1})$  unter Verwendung von  $2^n$  Logikfunktionssignalen  $c_0, c_1, \dots, c_{2^n-1}$  mit  $c_i \in \{1, 0\}$  zum Vorgeben der Logikfunktion gemäß folgender Abbildungsvorschrift:

$$y(a_{n-1}, \dots, a_1, a_0) = \overline{a_{n-1}} \dots \overline{a_2} \cdot y_0 \vee \overline{a_{n-1}} \dots a_2 \cdot y_1 \vee \dots \vee a_{n-1} \dots a_2 \cdot y_{2^{n-2}-1}$$

mit

$$y_0 = \overline{c_0} \cdot \overline{a_1} \cdot \overline{a_0} \vee \overline{c_1} \cdot \overline{a_1} \cdot a_0 \vee \overline{c_2} \cdot a_1 \cdot \overline{a_0} \vee \overline{c_3} \cdot a_1 \cdot a_0$$

und

$$\overline{y_0} = c_0 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_1 \cdot \overline{a_1} \cdot a_0 \vee c_2 \cdot a_1 \cdot \overline{a_0} \vee c_3 \cdot a_1 \cdot a_0$$

und

$$y_1 = \overline{c_4} \cdot \overline{a_1} \cdot \overline{a_0} \vee \overline{c_5} \cdot \overline{a_1} \cdot a_0 \vee \overline{c_6} \cdot a_1 \cdot \overline{a_0} \vee \overline{c_7} \cdot a_1 \cdot a_0$$

und

$$\overline{y_1} = c_4 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_5 \cdot \overline{a_1} \cdot a_0 \vee c_6 \cdot \overline{a_1} \cdot \overline{a_0} \vee c_7 \cdot \overline{a_1} \cdot \overline{a_0}$$

...

und

$$y_{2^{(n-2)}-1} = c_{2^{n-4}} \cdot \overline{a_1} \cdot \overline{a_0} \vee c_{2^{n-3}} \cdot \overline{a_1} \cdot a_0 \vee c_{2^{n-2}} \cdot \overline{a_1} \cdot \overline{a_0} \vee c_{2^{n-1}} \cdot a_1 \cdot a_0$$

und

$$y_{2^{(n-2)}-1} = c_{2^{n-4}} \cdot \overline{a_1} \cdot \overline{a_0} \vee c_{2^{n-3}} \cdot \overline{a_1} \cdot \overline{a_0} \vee c_{2^{n-2}} \cdot \overline{a_1} \cdot \overline{a_0} \vee c_{2^{n-1}} \cdot \overline{a_1} \cdot \overline{a_0}$$

22. Logik-Schaltkreis-Anordnung nach einem der Ansprüche 1 bis 18, bei der die Transistoren verschaltet sind zum Verarbeiten von n Datensignalen  $a_0, a_1, \dots, a_{n-1}$  zu einem Ausgangssignal  $y(a_0, a_1, \dots, a_{n-1})$  unter Verwendung von Logikfunktionssignalen  $c_0, c_1, \dots, c_{2^{(n-1)}}$  mit  $c_i \in \{1, 0\}$  zum Vorgeben der Logikfunktion gemäß folgender Abbildungsvorschrift:

$$y = c_0 \cdot \overline{a_{n-1}} \cdot \dots \cdot \overline{a_1} \cdot \overline{a_0} \vee \dots \vee c_{2^{n-2}} \cdot a_{n-1} \cdot \dots \cdot a_1 \cdot \overline{a_0} \vee c_{2^{n-1}} \cdot a_{n-1} \cdot \dots \cdot a_1 \cdot a_0$$

23. Logik-Schaltkreis-Anordnung nach einem der Ansprüche 11 bis 22, bei der die Transistoren der Signalfade derart verschaltet sind, dass einzig mittels Vorgebens des mindestens einen Logikfunktionssignals eine zugeordnete der Logikfunktionen eindeutig vorgegeben ist.

Es folgen 7 Blatt Zeichnungen

FIG 1

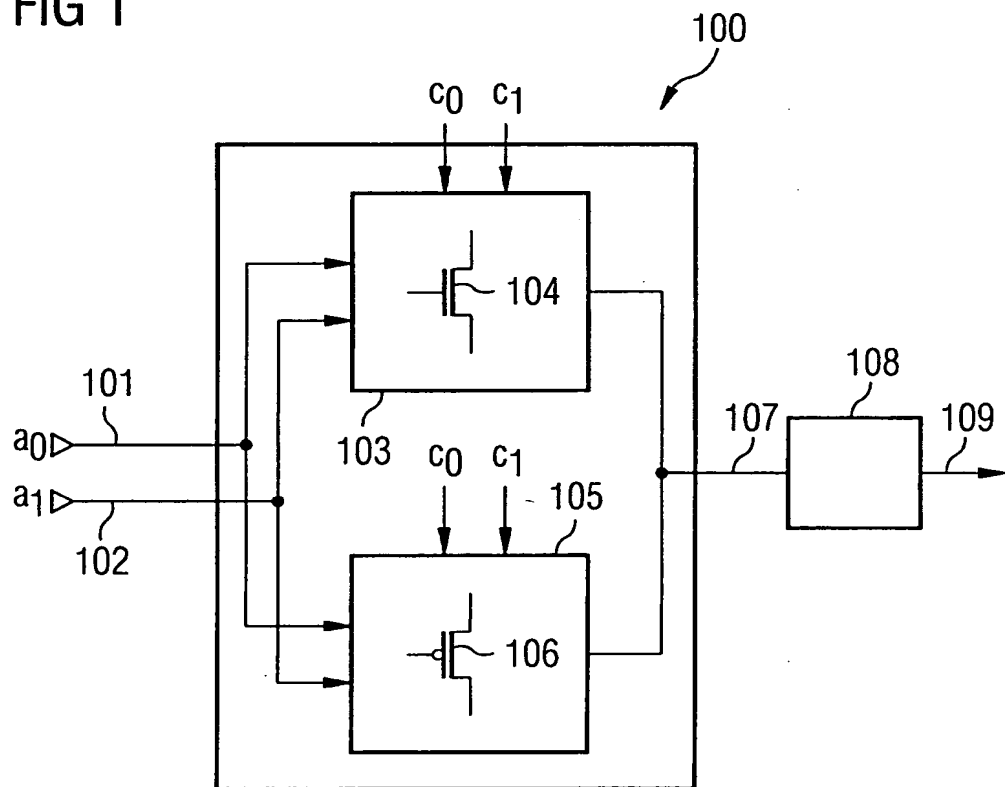


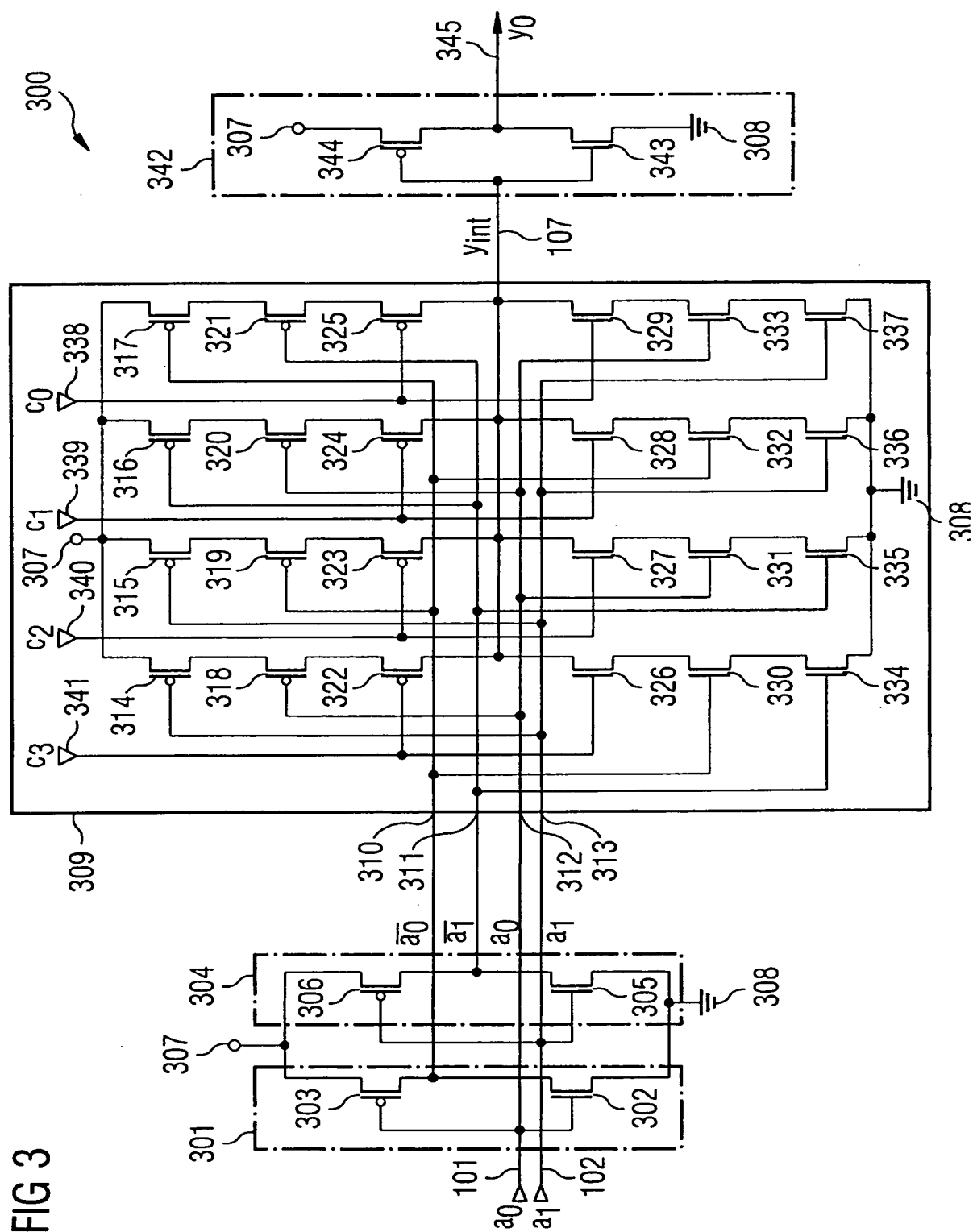


FIG 2

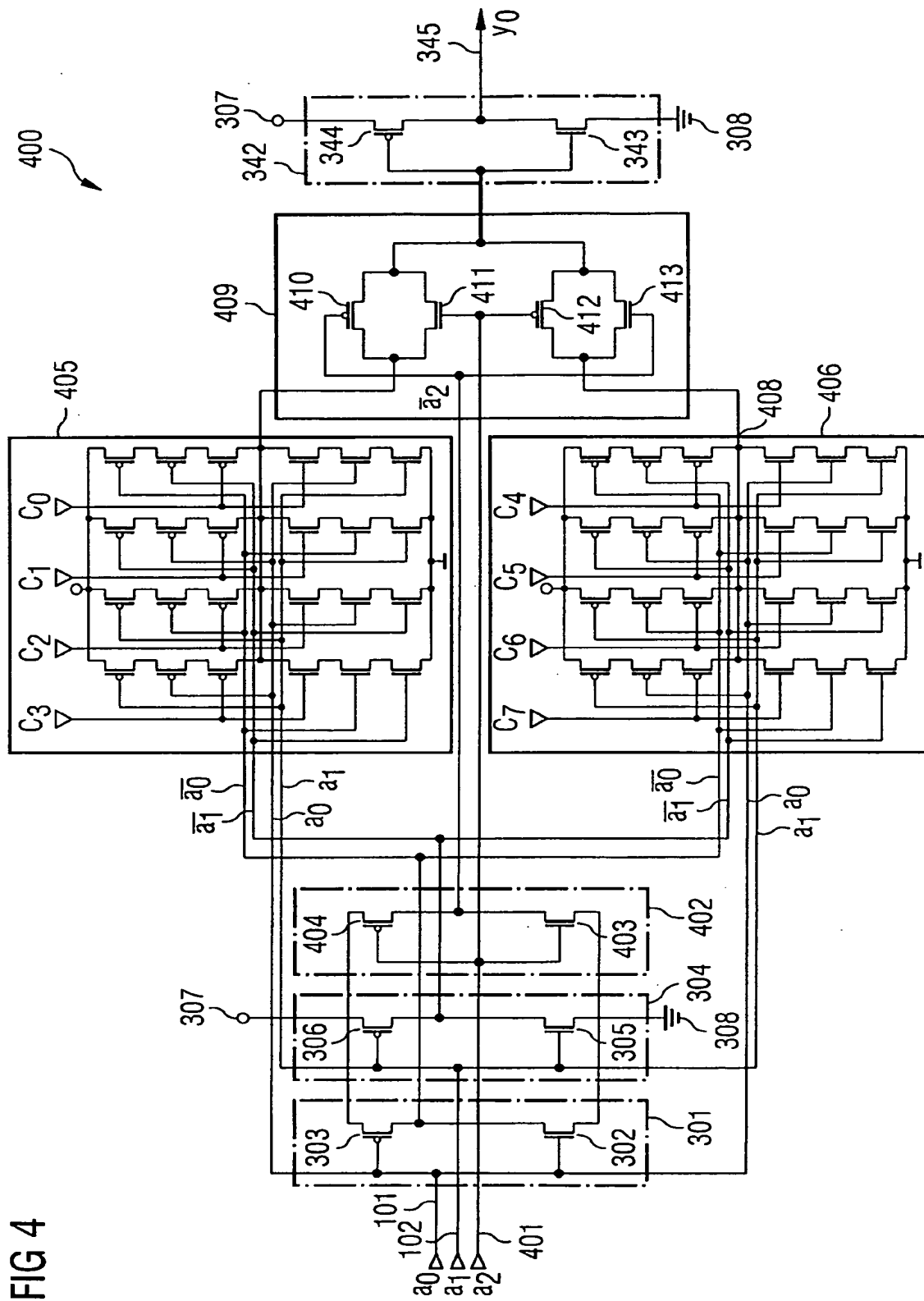
200

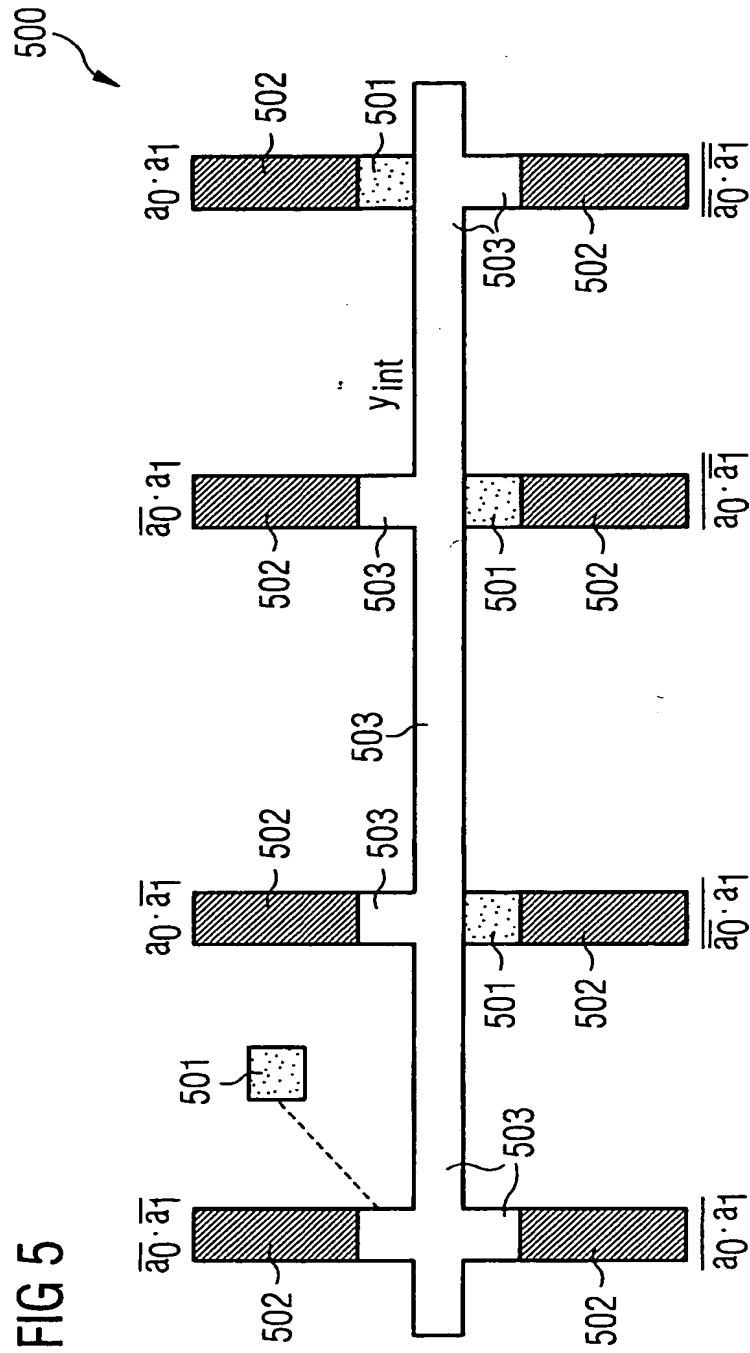
c <sub>3</sub>	c <sub>2</sub>	c <sub>1</sub>	c <sub>0</sub>	y
0	0	0	0	0
0	0	0	1	$a_0 \wedge a_1$ [AND]
0	0	1	0	$\overline{a_0} \wedge a_1$
0	0	1	1	$a_1$
0	1	0	0	$a_0 \wedge \overline{a_1}$
0	1	0	1	$a_0$
0	1	1	0	$a_0 \oplus a_1$ [XOR]
0	1	1	1	$a_0 \vee a_1$ [OR]
1	0	0	0	$\overline{a_0 \vee a_1}$ [NOR]
1	0	0	1	$\overline{a_0 \oplus a_1}$ [XNOR]
1	0	1	0	$\overline{a_0}$
1	0	1	1	$\overline{a_0} \vee a_1$
1	1	0	0	$\overline{a_1}$
1	1	0	1	$a_0 \vee \overline{a_1}$
1	1	1	0	$\overline{a_0 \wedge a_1}$ [NAND]
1	1	1	1	1

FIG 3



**FIG 4**





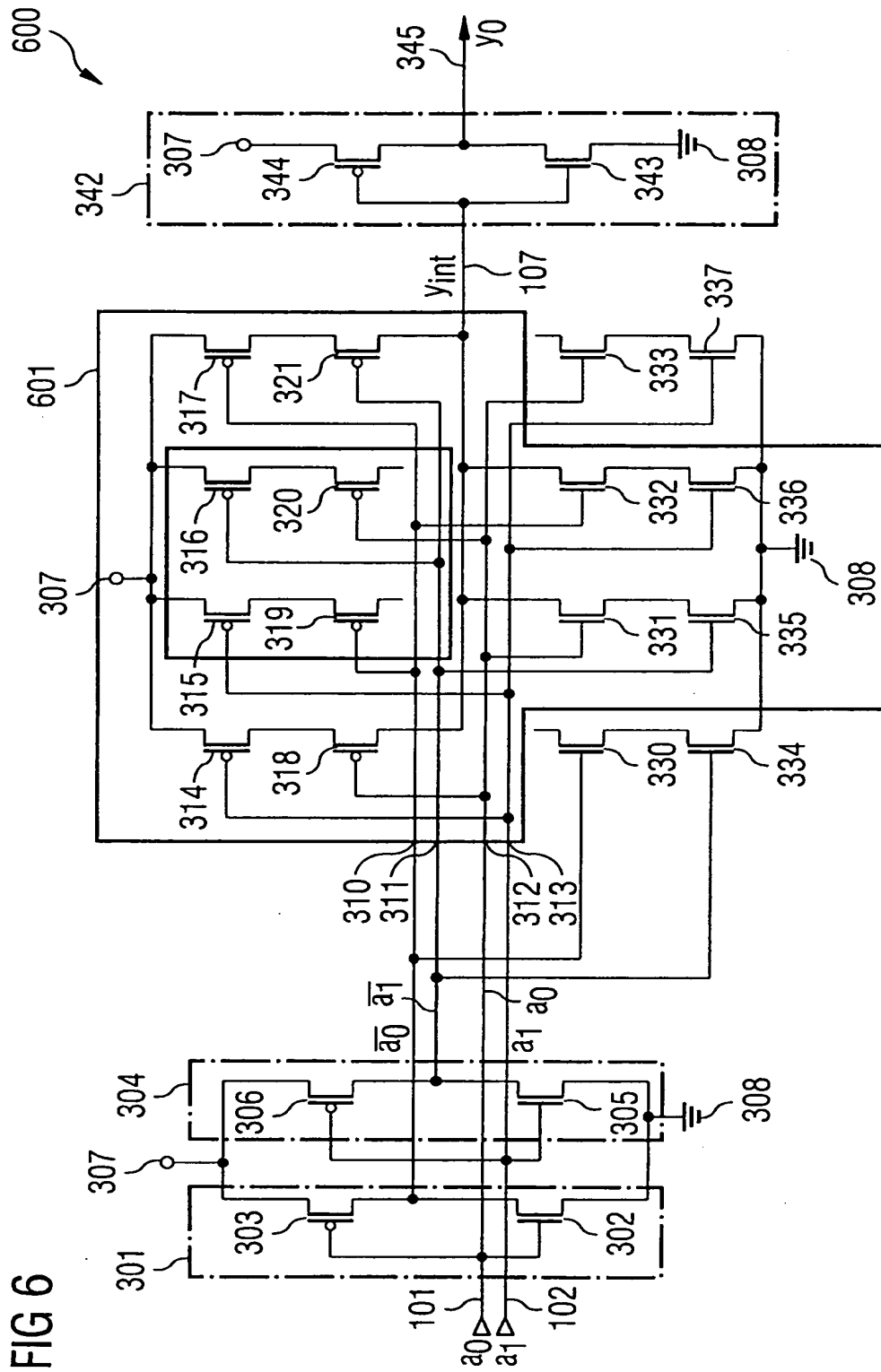


FIG 7

